

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 1



**Sistema de Gestión de Incidencias para la
Universidad de las Ciencias Informáticas.**

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

AUTORES: Migdelys Sablón Loforte

Manuel Alejandro Quert Gómez

TUTOR: Ing. Roberlán Rodríguez Sánchez

Ciudad de la Habana, 18 de junio del 2009

“Año del 50 Aniversario del Triunfo de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores del trabajo titulado: “Sistema de Gestión de Incidencias para la Universidad de las Ciencias Informáticas” y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Firma de la autora: Migdelys Sablón Loforte.

Firma del autor: Manuel Alejandro Quert Gómez.

Firma del tutor: Ing. Roberlán Rodríguez Sánchez

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

OPINIÓN DEL CLIENTE

Agradecimientos:

A mis padres y a mi hermana, por confiar en mí y por apoyarme siempre.

A mi compañero de tesis Manuel, por todas las fuerzas que me diste para seguir adelante y por la paciencia que tuviste conmigo.

A mi tutor Roberlán, por su entrega incondicional a este trabajo.

A mi novio, por estar a mi lado todos estos años.

A mi suegra, por brindarme su ayuda y su confianza.

A los profesores que contribuyeron con el desarrollo de este trabajo.

A mis compañeros y amigos, gracias por brindarme su apoyo.

A todos, muchas gracias.

Migdelys

Agradecimientos:

A mi familia, que no me dejó preocuparme por otra cosa que no fuera estudiar.

A mis amigos, por estar siempre que los necesitaba.

A mi tutor Roberlán, por apoyarnos durante dos años de proyecto productivo.

A Julio César y Jose Miguel, por su ENORME paciencia.

Manuel Alejandro

Dedicatoria:

A mis padres.

Migdelys

Dedicatoria:

A mi familia.

A mis amigos.

Manuel Alejandro

RESUMEN:

En la Universidad de las Ciencias Informáticas diariamente ocurren numerosas incidencias. Llevar el control de las mismas se hace muy difícil, debido a que el área territorial que ocupa la universidad es muy extensa. En la Oficina de Seguridad y Protección se desarrollan diversos procesos manualmente, entre ellos el trabajo con las incidencias. En la actualidad cuando se detecta una incidencia se informa por correo al Especialista de Seguridad y Protección, quién es el encargado de investigarlas y crear el expediente de cada una de ellas. El expediente es una planilla en la que se escriben todos los datos de la incidencia. Se realizan búsquedas de expedientes o de personas para saber si estuvieron implicadas en alguna incidencia y se elaboran reportes e informes. La pérdida de información, el trabajo manual que se realiza con los expedientes y la gran cantidad de archivos físicos que existen, son las dificultades más grandes presentes en los procesos de gestión de incidencias. El Sistema de Gestión de Incidencias para la Universidad de las Ciencias Informáticas surge por la necesidad que tiene la universidad de automatizar dichos procesos. Este aportará beneficios como: agilizar el desarrollo de los procesos de gestión de incidencias, mejorar la auditabilidad de la información, ya que la misma se encontrará digitalizada en la base de datos del sistema, obtener una reducción de los gastos producto del consumo de material de oficina, disminuir los riesgos por pérdida y deterioro de la información y aumentar el nivel de seguridad de la misma.

Palabras clave: sistema, gestión, incidencias.

ÍNDICE

INTRODUCCIÓN.....	15
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....	19
1.1 Introducción	19
1.2 Sistemas Similares.....	19
1.2.1 Sistemas similares en la Universidad	21
1.2.2 ¿Por qué hacer uno, si existen otros?	21
1.3 Sistema de Gestión de Contenido (CMS)	22
1.4 Lenguaje de programación para páginas dinámicas.....	23
1.4.1 PHP 5.....	24
1.5 Sistemas Gestores de Bases de Datos (SGBD).....	24
1.6 Entorno de Desarrollo Integrado (IDE).....	26
1.7 Servidor de aplicaciones.....	27
1.8 Metodologías de desarrollo de software.....	27
1.8.1 Metodologías robustas.....	28
1.8.2 Metodologías ágiles.....	28
1.8.3 ¿Por qué Scrum y XP?	33
1.8.4 Unified Modeling Language (UML).....	33
1.8.5 Visual Paradigm	34
1.9 Conclusiones	34
CAPÍTULO 2. PROPUESTA DE SOLUCIÓN.....	35
2.1 Introducción	35
2.2 Descripción de los procesos de gestión de incidencias	35
2.3 Propósito del sistema.....	36
2.4 Descripción del sistema propuesto	37
2.4.1 Roles de usuarios.....	37
2.4.2 Arquitectura y diseño	38
2.4.3 Despliegue.....	40
2.4.4 Seguridad del Sistema	41
2.4.5 Product Backlog	42
2.4.6 Requerimientos adicionales.....	49
2.4.7 Planificación de los Sprints	50
2.5 Construcción de la propuesta de solución	69
2.5.1 Diagramas de pantallas por funcionalidades	69
2.5.2 Tratamiento de Excepciones	78
2.5.3 Estándares de codificación	78
2.6 Conclusiones	80
CAPÍTULO 3. PRUEBAS DEL SISTEMA.....	81
3.1 Introducción	81
3.2 Modelo de Pruebas	81
3.2.1 Casos de Prueba por historias de usuario	81

3.3 Conclusiones	99
CONCLUSIONES	100
RECOMENDACIONES.....	101
REFERENCIAS BIBLIOGRÁFICAS	102
BIBLIOGRAFÍA	104
GLOSARIO DE TÉRMINOS	105
ANEXOS	107

ÍNDICE DE TABLAS

Tabla 2.1 Definición de los roles de usuarios.....	37
Tabla 2.2 Historia de usuario Autenticar Usuarios.....	42
Tabla 2.3 Historia de usuario Administrar Usuarios.....	42
Tabla 2.4 Historia de usuario Buscar Contenido.....	43
Tabla 2.5 Historia de usuario Mostrar Contenido.....	43
Tabla 2.6 Historia de usuario Reportar una Incidencia.....	43
Tabla 2.7 Historia de usuario Modificar una Incidencia.....	44
Tabla 2.8 Historia de usuario Eliminar una Incidencia.....	44
Tabla 2.9 Historia de usuario Crear un Expediente.....	44
Tabla 2.10 Historia de usuario Modificar un Expediente.....	45
Tabla 2.11 Historia de usuario Eliminar un Expediente.....	45
Tabla 2.12 Historia de usuario Adicionar un Infractor.....	46
Tabla 2.13 Historia de usuario Modificar un Infractor.....	46
Tabla 2.14 Historia de usuario Eliminar un Infractor.....	46
Tabla 2.15 Historia de usuario Crear un Reporte Semanal.....	47
Tabla 2.16 Historia de usuario Modificar un Reporte Semanal.....	47
Tabla 2.17 Historia de usuario Eliminar un Reporte Semanal.....	47
Tabla 2.18 Historia de usuario Crear un Informe Mensual.....	48
Tabla 2.19 Historia de usuario Modificar un Informe Mensual.....	48
Tabla 2.20 Historia de usuario Eliminar un Informe Mensual.....	48
Tabla 2.21 Historia técnica Diseñar interfaz.....	49
Tabla 2.22 Historia técnica Diseñar la ayuda del sistema.....	49
Tabla 2.23 Tarea Crear formulario Reportar Incidencia.....	52
Tabla 2.24 Tarea Validar campos numéricos y obligatorios del formulario Reportar Incidencia.....	52
Tabla 2.25 Tarea Elaborar formulario Crear Expediente.....	52
Tabla 2.26 Tarea Validar campos obligatorios y numéricos del formulario Crear Expediente.....	53
Tabla 2.27 Tarea Crear formulario Adicionar Infractor.....	53
Tabla 2.28 Tarea Validar campos numéricos y obligatorios del formulario Adicionar Infractor.....	53
Tabla 2.29 Tarea Elaborar formulario Crear Reporte Semanal.....	54
Tabla 2.30 Tarea Validar campos obligatorios del formulario Crear Reporte Semanal.....	54
Tabla 2.31 Tarea Elaborar formulario Crear Informe Mensual.....	54
Tabla 2.32 Tarea Validar campos obligatorios del formulario Crear Informe Mensual.....	55
Tabla 2.33 Tarea Crear buscador de Incidencias.....	57
Tabla 2.34 Tarea Crear buscador de Expedientes.....	57
Tabla 2.35 Tarea Crear buscador de Infractores.....	58
Tabla 2.36 Tarea Crear buscador de Reportes Semanales.....	58
Tabla 2.37 Tarea Crear buscador de Informes Mensuales.....	58
Tabla 2.38 Tarea Crear enlace Editar a los formularios de las Incidencias.....	59
Tabla 2.39 Tarea Crear enlace Editar a los formularios de los Expedientes.....	59
Tabla 2.40 Tarea Crear enlace Modificar a los formularios de los Infractores.....	59
Tabla 2.41 Tarea Crear enlace Editar a los formularios de los Reportes Semanales.....	60
Tabla 2.42 Tarea Crear enlace Editar a los formularios de los Informes Mensuales.....	60
Tabla 2.43 Tarea Crear enlace Borrar a los formularios de las Incidencias.....	60
Tabla 2.44 Tarea Crear enlace Borrar a los formularios de los Expedientes.....	61

Tabla 2.45 Tarea Crear enlace Eliminar a los formularios de los Infractores.....	61
Tabla 2.46 Tarea Crear enlace Borrar a los formularios de los Reportes Semanales.....	61
Tabla 2.47 Tarea Crear enlace Borrar a los formularios de los Informes Mensuales.....	62
Tabla 2.48 Tarea Implementar la vista de las incidencias en formato PDF.....	63
Tabla 2.49 Tarea Implementar la vista de los expedientes en formato PDF.....	64
Tabla 2.50 Tarea Implementar la vista de los Infractores en formato PDF.....	64
Tabla 2.51 Tarea Implementar la vista de los Reportes Semanales en formato PDF.....	64
Tabla 2.52 Tarea Implementar la vista de los Informes Mensuales en formato PDF.....	65
Tabla 2.53 Tarea Configurar la autenticación en el sistema.....	66
Tabla 2.54 Tarea Crear y configurar el rol Técnico.....	67
Tabla 2.55 Tarea Crear y configurar el rol Especialista.....	67
Tabla 2.56 Tarea Crear menú principal.....	67
Tabla 2.57 Tarea Seleccionar y configurar la interfaz.....	68
Tabla 2.58 Tarea Configuración e implementación de la ayuda.....	68
Tabla 3.1 Caso de prueba Autenticarse correctamente.....	81
Tabla 3.2 Caso de prueba Autenticarse incorrectamente.....	82
Tabla 3.3 Caso de prueba Control de acceso de usuarios no autorizados.....	82
Tabla 3.4 Caso de prueba Control de acceso a la información.....	82
Tabla 3.5 Caso de prueba Mostrar el contenido de una Incidencia.....	83
Tabla 3.6 Caso de prueba Mostrar el contenido de un Expediente.....	83
Tabla 3.7 Caso de prueba Mostrar el contenido de un Infractor.....	84
Tabla 3.8 Caso de prueba Mostrar el contenido de un Informe Mensual.....	84
Tabla 3.9 Caso de prueba Mostrar el contenido de un Reporte Semanal.....	85
Tabla 3.10 Caso de prueba Validación de los campos numéricos del formulario Reportar Incidencia.....	85
Tabla 3.11 Caso de prueba Validación de los campos obligatorios del formulario Reportar Incidencia.....	85
Tabla 3.12 Caso de prueba Reportar una Incidencia.....	86
Tabla 3.13 Caso de prueba Modificar datos de una Incidencia.....	87
Tabla 3.14 Caso de prueba Eliminar una Incidencia.....	87
Tabla 3.15 Caso de prueba Validación de los campos numéricos del formulario Crear Expediente.....	88
Tabla 3.16 Caso de prueba Validación de los campos obligatorios del formulario Crear Expediente.....	88
Tabla 3.17 Caso de prueba Crear un Expediente.....	89
Tabla 3.18 Caso de prueba Modificar un Expediente.....	90
Tabla 3.19 Caso de prueba Eliminar un Expediente.....	90
Tabla 3.20 Caso de prueba Adicionar un Infractor.....	91
Tabla 3.21 Caso de prueba Validación de los campos obligatorios del formulario Adicionar Infractor.....	91
Tabla 3.22 Caso de prueba Validación de los campos numéricos del formulario Adicionar Infractor.....	92
Tabla 3.23 Caso de prueba Modificar los datos de un Infractor.....	92
Tabla 3.24 Caso de prueba Eliminar un Infractor.....	93
Tabla 3.25 Caso de prueba Validación de los campos obligatorios del formulario Crear Reporte Semanal.....	93
Tabla 3.26 Caso de prueba Crear un Reporte Semanal.....	94
Tabla 3.27 Caso de prueba Modificar datos de un Reporte Semanal.....	94
Tabla 3.28 Caso de prueba Eliminar un Reporte Semanal.....	95
Tabla 3.29 Caso de prueba Validación de los campos obligatorios del formulario Crear un Informe Mensual.....	95

Tabla 3.30 Caso de prueba Crear un Informe Mensual.....	96
Tabla 3.31 Caso de prueba Modificar datos de un Informe Mensual.	96
Tabla 3.32 Caso de prueba Eliminar un Informe Mensual.....	97
Tabla 3.33 Caso de prueba Buscar una Incidencia.	97
Tabla 3.34 Caso de prueba Buscar un Expediente.....	98
Tabla 3.35 Caso de prueba Buscar un Infractor.	98
Tabla 3.36 Caso de prueba Buscar un Reporte Semanal.	99
Tabla 3.37 Caso de prueba Buscar Contenido.	99

ÍNDICE DE FIGURAS

Figura 1. Diagrama de despliegue.....41
Figura 2. Diagrama Burndown chart *Sprint* 1.56
Figura 3. Diagrama Burndown chart *Sprint* 2.62
Figura 4. Diagrama Burndown chart *Sprint* 3.65
Figura 5. Diagrama Burndown chart *Sprint* 4.69
Figura 6. Diagrama de pantalla Reportar Incidencia.....70
Figura 7. Diagrama de pantalla Crear Reporte Semanal.....70
Figura 8. Diagrama de pantalla Crear Informe Mensual.70
Figura 9. Diagrama de pantalla Crear Expediente.71
Figura 10. Diagrama de pantalla Buscar Incidencias.....72
Figura 11. Diagrama de pantalla Buscar Expedientes.....73
Figura 12. Diagrama de pantalla Adicionar Infractor.....74
Figura 13. Diagrama de pantalla Buscar Infractor.....75
Figura 14. Diagrama de pantalla Buscar Reporte Semanal.....76
Figura 15. Diagrama de pantalla Buscar Informe Mensual.....77

INTRODUCCIÓN

Actualmente todas las empresas del mundo tienen como prioridad la seguridad y protección de sus recursos. Razón por la cual dedican parte de sus presupuestos para implementar mejores estrategias de seguridad, con el fin de protegerse de los distintos ataques y amenazas que puedan atentar contra la seguridad de sus medios.

La Universidad de las Ciencias Informáticas (UCI) ocupa un área extensa territorialmente y cuenta con una gran cantidad de recursos materiales a disposición de todas las personas que trabajan o estudian en la misma. Debido a esto surge el Sistema de Seguridad y Protección, un sistema organizacional, que se encarga del control de acceso, del cuidado de medios materiales y de la disciplina interna de la universidad.

El robo de una computadora, la propaganda contrarrevolucionaria o cualquier suceso que atente contra la seguridad de los medios o de las personas que permanecen en la universidad, son incidencias que forman parte del grupo de acciones que violan el Sistema de Seguridad y Protección. Estos incidentes deben ser informados y registrados en la Oficina de Seguridad y Protección.

En esta oficina se desarrollan numerosos procesos manualmente, pero el que más información genera es el registro de incidencias, por lo que se hace muy difícil llevar a cabo la contabilidad de las mismas y hasta el momento no existe ningún sistema informático capaz de tramitar con eficiencia toda la información que se genera diariamente. Partiendo de estos hechos se define la siguiente **situación problémica**, a la cual es necesario dar solución:

Verificar si una persona ha cometido una incidencia, buscar información contenida en los expedientes para consultarla o modificarla y elaborar los reportes son tareas manuales, repetitivas y toman mucho tiempo debido a la gran cantidad de archivos físicos que existen. Es tanta la información acumulada con la cual se trabaja diariamente, que en ocasiones ocurren pérdidas de algunos documentos o deterioro de los mismos, lo que provoca la disminución de la credibilidad de los datos. Los informes pueden presentar errores debido a que los cálculos que se realiza son manuales, lo cual es responsabilidad del Especialista de Seguridad y Protección de la UCI, que es el encargado de trabajar con toda esta información.

Por lo antes planteado se tiene el siguiente **problema científico**: ¿Cómo digitalizar los procesos de gestión de incidencias en la Oficina de Seguridad y Protección en la UCI?

Para darle solución a los problemas existentes y debido a la necesidad que tiene la universidad de ejercer un control más eficiente sobre las incidencias, se decidió desarrollar un sistema Web para automatizar los procesos de gestión de incidencias. Surge así, el Sistema de Gestión de Incidencias para la UCI.

El sistema facilitará el trabajo del Especialista de Seguridad y Protección, pero lo más importante es poder contar con un sistema seguro y confiable para almacenar la información. Se brindará un servicio rápido a los usuarios del sistema, se podrán realizar búsquedas teniendo en cuenta algunos criterios y generar e imprimir reportes cada determinado intervalo de tiempo con rapidez y eficiencia, ya que la información se encontrará digitalizada en la base de datos del sistema. El Sistema de Gestión de Incidencias para la UCI contendrá todos los datos concernientes a las incidencias ocurridas en la universidad.

Por tanto el **objeto de estudio** de este trabajo son los procesos de gestión de incidencias que ocurren en la UCI.

El **campo de acción** queda enmarcado en la Oficina de Seguridad y Protección y el Puesto de Mando de la UCI donde se llevan a cabo los procesos que se tendrán en cuenta para la realización del sistema.

Esto hace plantear la siguiente **idea a defender**: Con la implementación de un sistema Web para la gestión de incidencias en la Oficina de Seguridad y Protección en la UCI se mejorará la gestión de dicha información.

El **objetivo general** que persigue el trabajo es desarrollar un sistema Web que garantice la gestión de incidencias en la UCI.

De aquí, se derivan los siguientes **objetivos específicos**:

- Diseñar el marco teórico.
- Identificar los procesos presentes en la gestión de incidencias en la UCI.
- Implementar el Sistema de Gestión de Incidencias para la UCI.
- Realizar pruebas de calidad al Sistema de Gestión de Incidencias para la UCI.

Para cumplir los objetivos trazados, se definieron las siguientes **tareas**:

- Realización del estado del arte de sistemas para gestionar incidencias a nivel internacional y de la UCI.
- Entrevista a clientes del sistema.
- Estudio de los procesos presentes en la gestión de las incidencias en la UCI.

- Especificación, validación y levantamiento de requisitos.
- Implementación de los script, para generar documentos en formato PDF.
- Preparación de los casos de prueba necesarios para comprobar las funcionalidades del sistema.

Los **métodos científicos** que se utilizan en la realización de este trabajo, permiten estudiar las características del objeto de investigación.

➤ **Métodos Teóricos**

- El método Analítico - Sintético facilitó el procesamiento de la información obtenida en la investigación realizada sobre el objeto de estudio.
- El método Análisis histórico – Lógico posibilitó el conocimiento del estado de arte del objeto de estudio, así como su evolución y desarrollo.

➤ **Métodos Empíricos**

- El método Observación proporcionó una visión realista y objetiva de los procesos, conceptos y definiciones presentes en el negocio.
- El método Entrevista permitió obtener la información necesaria relacionada con los problemas presentes en los procesos de gestión de incidencias, mediante entrevistas realizadas a personas que trabajan en la Oficina de Seguridad y Protección y en el Puesto de Mando UCI, específicamente al Especialista de Seguridad y Protección y a los Oficiales Operativos del Puesto de Mando UCI.

Estructuración del contenido

El contenido de este trabajo está distribuido por capítulos, los que a su vez están distribuidos por temáticas. El informe cuenta con tres capítulos de los cuales se describe de forma general su contenido:

Capítulo I

En este capítulo se describe el estado del arte de sistemas que gestionan información de incidencias. Se enuncian y describen las metodologías, el lenguaje de programación y las herramientas en las que se decidió desarrollar el sistema.

Capítulo II

En este capítulo se define la propuesta de solución del trabajo. Se precisan y describen los procesos de gestión de incidencias. Se enuncian los requisitos que debe cumplir la aplicación. Se describe la arquitectura y los patrones de diseño más utilizado en el desarrollo del sistema, así como la forma en que se previenen los ataques de seguridad al mismo.

Capítulo III

En este capítulo se presentan los casos de prueba necesarios para comprobar la calidad del sistema.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

En este capítulo se abordarán varios temas, en los que se planteará la necesidad de implementar un sistema para mejorar la gestión de las incidencias ocurridas en la UCI. Se realizará un estudio sobre los sistemas existentes que gestionan información sobre incidencias y sobre las distintas herramientas que se utilizan en el diseño e implementación de este tipo de aplicaciones. Se concluye justificando la selección de las herramientas, metodologías y lenguajes con las que se elaborará el trabajo.

1.2 Sistemas Similares

Se llama **incidencia** a cualquier hecho delictivo o indisciplina que ocurra dentro de la universidad y que atente contra la seguridad de las personas o medios que se encuentran en la misma.

Se denominan **infractor** a la persona que comete una incidencia.

Un **expediente** es un documento en el que se recoge toda la información obtenida durante la investigación de la incidencia.

El **reporte semanal** y el **informe mensual** son reportes que detallan el comportamiento de las incidencias en un período de tiempo determinado. (Ver anexo 1)

La **gestión de la información** es un proceso que incluye operaciones como extracción, manipulación, tratamiento, depuración, conservación, acceso y/o colaboración de la información adquirida por una organización a través de diferentes fuentes y que gestiona el acceso y los derechos de los usuarios sobre la misma. [1]

Los **sistemas Web**, son las aplicaciones a las que los usuarios tienen acceso mediante un servidor Web a través de una intranet o Internet utilizando un navegador.

Un **sistema Web de gestión de incidencias** es un conjunto de procesos automatizados que se ejecutan sobre una plataforma digital, orientados a manipular información relacionada con las incidencias ocurridas en una entidad determinada.

Dichos sistemas contribuyen a que exista una mejor comunicación entre los usuarios y la información. Permiten que el cliente acceda a los datos rápidamente. Brindan privacidad en el acceso, ya que al sistema solo pueden acceder aquellas personas que están autorizadas a trabajar con dicha información, en este caso el Especialista de Seguridad y Protección y los Oficiales Operativos. Los sistemas

informáticos brindan la posibilidad de gestionar cualquier tipo de información, de manera rápida, eficiente y confiable.

Con la implementación de un sistema para la gestión de incidencias, la información podrá guardarse en una base de datos que permitirá que la información pueda ser consultada desde cualquier ordenador autorizado y con acceso a la red de la universidad. Se podrá almacenar y manejar de manera clara, sencilla y ordenada, toda la información relacionada con los procesos de gestión de incidencias. Evitando así las pérdidas o deterioro de los documentos, debido a la mala calidad del papel en el se imprimen o porque simplemente son muy antiguos debido a que se almacenan durante tres años.

Hoy en día cada país, incluso cada provincia, estado, municipio o dependencia, busca como informatizar el trabajo de sus fuerzas de seguridad. Desde sistemas simples de almacenamiento de datos, hasta complejos sistemas que integran reportes, procesamiento de información e integración con otros sistemas relacionados.

Todas las empresas que cuentan con algún sistema de gestión de incidencias, tienen la posibilidad de conocer los datos relativos a los hechos delictivos ocurridos dentro de la entidad, entre ellos las personas implicadas, que hicieron, como lo hicieron, cuando y donde. Esto les brinda la oportunidad de prevenir algunos de esos sucesos, debido a que gracias a toda la información almacenada, se podrán sacar estadísticas que permitirán determinar el lugar de más incidencias delictivas, tipos de delitos efectuados con mayor frecuencia, edades de las personas que más incurrir en los hechos, entre otras cosas. Dada la importancia y la utilidad de estos sistemas, cada día son más las instituciones, localidades o países, que se preocupan por crearlos con el fin de disminuir los delitos que se cometen diariamente.

En Berlín, la capital de Alemania, se ha tenido un gran éxito en la disminución de la tasa de criminalidad gracias al sistema POLIKS. Este sistema fue diseñado y ejecutado por T-Systems, sobre la base de los requisitos de las fuerzas de la Policía de Berlín y se puso en marcha en marzo del 2005. T-Systems es una compañía que se dedica a prestar servicios de integración de sistemas, creación de plataformas y software para la seguridad empresarial y estatal.

POLIKS ofrece una amplia gama de características, tales como: el comercio electrónico y gestión de procesos de adquisición de datos, un sistema de búsqueda para la localización de personas y objetos desaparecidos, un sistema de recuperación de información, un sistema de estadísticas que establece las relaciones entre policías y delitos y un sistema electrónico de gestión de los registros de antecedentes penales. [2]

Entre los sistemas que gestionan información relacionada con delitos, se encuentra el sistema I-24/7 perteneciente a la Organización Internacional de Policía Criminal (INTERPOL). La INTERPOL centraliza todos los datos relacionados con los hechos delictivos de interés para sus países miembros. Cuenta con una de las bases de datos más conocidas en el mundo, debido al alcance de la organización y la gran cantidad de información que almacena.

EL sistema I-24/7 se puso en marcha en el 2003. Ofrece servicios de comunicación permanente y tiene como propósito revolucionar el modo de intercambiar información y realizar investigaciones internacionales. Ha superado todas las expectativas en términos de utilización y eficiencia, gracias a la acción coordinada de las fuerzas de policía de los países miembros de la INTERPOL, que a la vez proporcionan y solicitan informaciones y servicios a este sistema. [3]

Sistemas similares al sistema I-24/7 han sido implementados en varios países. Uno de esos países es Argentina. En la Provincia de Mendoza, se desarrolla el Plan de Identificación Biométrica; el cual centra su funcionamiento en un sistema de identificación que dispone de una base de datos biométrica, a fin de poder identificar unívocamente a cada uno de los ciudadanos de esa provincia.

El Sistema de Identificación incluye tres módulos: Sistema del Registro Civil, Sistema de la Policía y Sistema Penitenciario. El Sistema de la Policía registra a todos los ciudadanos que han cometido algún hecho delictivo tras haberle tomado la impresión digital de sus dedos. Esto permite tener constancia de que esa persona tiene antecedentes. El sistema brinda velocidad de identificación, mejora la eficiencia en la resolución de hechos delictivos y admite el intercambio de información sobre el tema entre distintos organismos. [4]

1.2.1 Sistemas similares en la Universidad

En la Universidad de la Ciencias Informáticas no existe un sistema que se encargue de tramitar las incidencias que en la institución tienen lugar, es por ello la necesidad de desarrollar este trabajo.

1.2.2 ¿Por qué hacer uno, si existen otros?

En la actualidad se encuentren disímiles sistemas para gestionar y procesar información relacionada con el tema de la seguridad y la protección. Para este trabajo se decidió que no se utilizaran sistemas que existan en el mundo como los antes mencionados, pues cada uno de ellos fue implementado para satisfacer necesidades propias de la institución a la que pertenecen.

El sistema que se propone tiene una baja complejidad respecto al almacenamiento de datos, no tiene vinculación con ningún sistema externo nacional o extranjero, lo que lo hace atípico respecto a la tendencia actual. Cabe aclarar que dicha tendencia se debe a que los usuarios finales son cuerpos armados o de alta seguridad y eso no es lo que se desea en este trabajo.

POLIKS es un sistema muy poderoso, en la actualidad son 17.000 los usuarios que trabajan con él; 300,000 transacciones son procesadas diariamente y la disponibilidad operacional de la central del sistema es superior al 99 por ciento. La concurrencia del sistema a implementar no va a ser mayor de 15 personas, por lo que su nivel de complejidad es mucho menor que el de POLIKS. [2]

El sistema I-24/7 perteneciente a la INTERPOL, es un sistema que tiene características más generales a las que se persiguen con el desarrollo del sistema al que converge este trabajo. El sistema I-24/7 registra las personas y las causas por los cuales se buscan; y el sistema a desarrollar esta centrado principalmente en el desarrollo de un expediente por incidencias que ocurran. No tiene interés en registrar causas de búsqueda, pues esa tarea corresponde a un organismo diferente.

El Sistema de Identificación de la Provincia de Mendoza está integrado por tres sistemas: El Sistema de la Policía de la Provincia de Mendoza en Argentina, el Sistema del Registro Civil y el Sistema Penitenciario. Esta dependencia implica que cada persona que cometa en un hecho delictivo ya va a estar registrada en el sistema, cambiándose solamente los datos pertinentes a la incursión en delitos. Se puede dar la situación de que la persona ya esté registrada en el sistema, debido a que fue incluida en el mismo mediante el Sistema del Registro Civil. Además utiliza una base de datos biométrica que no es lo que necesita el sistema a desarrollar.

1.3 Sistema de Gestión de Contenido (CMS)

Un CMS trabaja con una o varias bases de datos que almacenan la información y están perfectamente separadas del diseño.

Drupal es un CMS modular muy configurable, de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Se puede extender o adaptar según las necesidades de cada cliente. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la Web y un énfasis especial en la usabilidad y consistencia de todo el sistema. El contenido puede ser tratado individualmente de la presentación si así el usuario lo desea. Cuenta con páginas de ayuda para los módulos del 'núcleo', tanto para usuarios como para administradores. Usa el

mod_rewrite de Apache para crear *URLs* (Localizador Uniforme de Recursos) que son manejables por los usuarios y los motores de búsqueda.

Para usarse en una intranet, Drupal se puede integrar con un servidor LDAP. Los administradores pueden asignar los permisos que crean necesarios a un rol determinado y agrupar los usuarios por roles. Su diseño es especialmente idóneo para construir y gestionar comunidades en Internet. Su flexibilidad y adaptabilidad, unidas a la gran cantidad de módulos adicionales que posee, hace que sea adecuado para realizar diferentes tipos de sitios Web. [5]

Puede trabajar con Apache o Microsoft IIS como servidor Web y en sistemas operativos como Linux, BSD, Solaris, Windows y Mac OS X. La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional. Toda la actividad y los sucesos del sistema se guardan en un 'registro de eventos', que puede ser visualizado por un administrador. Incluye un mecanismo de control de congestión, que permite habilitar y deshabilitar determinados módulos dependiendo de la carga del servidor. Tiene un mecanismo de caché que permite eliminar consultas a la base de datos, incrementando el rendimiento y reduciendo la carga del servidor.

La interfaz de usuario en Drupal se controla con el uso de temas. Un tema es un grupo de ficheros con información de diseño como CSS, XML y otras, que el núcleo o *Core* de Drupal interpreta y en base a ello configura la información generada. El sistema de temas separa el contenido de la presentación, permitiendo modificar el aspecto del sitio Web. Se pueden crear plantillas con HTML y/o con PHP. Por las razones antes expuestas se resolvió utilizar Drupal para la construcción de la aplicación.

1.4 Lenguaje de programación para páginas dinámicas

Se define como páginas dinámicas, a aquellas cuyo contenido se crea a partir de lo que un usuario introduce en una Web o formulario. Las aplicaciones más conocidas de las páginas Web dinámicas son:

- Mostrar información contenida en una base de datos.
- Modificar el contenido de una base de datos.
- Generar páginas Web de contenido estático.
- Mejorar la interacción entre el usuario y el sitio Web.

Existen diversos lenguajes que se utilizan en la creación de las páginas Web dinámicas, pero Personal Home Page (PHP) es el más popular y extendido en la Web, por el alto nivel que posee. [6]

1.4.1 PHP 5

PHP es un lenguaje muy potente, mundialmente utilizado en la programación de aplicaciones Web del lado del servidor. Es similar a otros lenguajes como C o C++, rápido, de sintaxis cómoda y su sencillez contribuye a su rápido aprendizaje. Sus primeras versiones, PHP 3 y PHP 4, sirvieron de gran ayuda a la comunidad de desarrolladores, para convertir a PHP en el lenguaje más utilizado en la construcción de páginas Web dinámicas.

Es un lenguaje interpretado, multiplataforma, se le pueden agregar extensiones fácilmente y dispone de una gran cantidad de librerías. Se puede utilizar como módulo de Apache, lo que lo hace extremadamente veloz. Por estar completamente escrito en C, se ejecuta rápidamente utilizando poca memoria. No hay que pagar licencias, no limita su distribución y se puede ampliar con nuevas funcionalidades si se desea. [7]

PHP 5 es la versión más adelantada, completa e intuitiva. El código es fácil de entender y mantener. Posee una vasta gama de funciones que le permiten adaptarse a cualquier entorno y a cualquier sistema operativo, por lo que es más eficiente.

El objetivo principal de PHP 5 es perfeccionar el lenguaje, utilizando los mecanismos de programación orientada a objetos (POO) que no estaban presentes en las versiones anteriores, parámetro necesario e imprescindible que convierte a PHP en un lenguaje idóneo para todo tipo de aplicaciones y entornos. Es un lenguaje muy asequible, tanto para aquellas personas que tienen experiencia en la programación de sistemas web dinámicos, como para los que no.

PHP unido al servidor web Apache, representan las dos herramientas claves para la creación de este tipo de aplicaciones. Su excelente forma de trabajar con casi todas las bases de datos, y en especial, con MySQL, con la que forma un equipo estrella dentro del mundo de desarrollo del software, hace de PHP el lenguaje perfecto para implementar el sistema de gestión de incidencias.

1.5 Sistemas Gestores de Bases de Datos (SGBD)

Un SGBD es un conjunto de programas que administran y gestionan la información contenida en una base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla y generar informes.

- Conservar la integridad de los datos.
- Controlar la seguridad y privacidad de los datos. [8]

Existen varios sistemas gestores de base de datos, entre los que han alcanzado mayor popularidad se encuentran PostgreSQL y MySQL, debido a las grandes funcionalidades con las que cuentan. Ambos SGBD cuentan con una base de usuarios bastante extendida, con multitud de interfaces, y documentación.

- **PostgreSQL** es un SGBD relacional orientada a objetos de software libre, de código abierto muy fiable, debido a que sus desarrolladores intentan liberar código estable y bien probado, que tenga la menor cantidad de errores posibles. Soporta una mayor carga de trabajo. Proporciona un sistema de bloqueo de tablas a nivel de fila, lo que admite realizar multitud de operaciones concurrentes sobre la misma tabla sin producir un retardo.

Provee un entorno más robusto, las cuestiones de rendimiento están siendo mejoradas rápidamente y en las últimas versiones los resultados son muy satisfactorios. Se puede ejecutar en cualquier plataforma moderna compatible con Unix. Corre sobre sistemas operativos basados en Microsoft Windows NT, tales como: Win2000, WinXP y Win2003. El consumo de recursos es muy elevado. En sus inicios era un poco más lento que MySQL, pero esto ha cambiado con la aparición de sus últimas versiones, que demuestran ser bastante compatibles, e incluso puede compararse con las bases de datos de código propietario. [9]

- **MySQL** es un sistema de administración de base de datos para base de datos relacionales, cuyo objetivo principal es la velocidad. Razón por la cual en un principio sus desarrolladores decidieron sacrificar ciertas prestaciones como: transacciones, llaves foráneas y *subselects*. A cambio se obtiene un mayor rendimiento.

Fue escrito en C y C++. Se adapta fácilmente a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl, Java y su integración en los diferentes sistemas operativos. Utiliza múltiples tablas para almacenar y organizar la información. Permite gestionar archivos llamados de bases de datos. Cuenta con una gran cantidad de herramientas que administran cómodamente la base de datos. En las últimas versiones se incorporaron algunas transacciones de las que se carecía al inicio.

Es muy destacable su condición de código abierto, de utilización gratuita, que puede modificarse con total libertad con solo descargar su código fuente. Esto ha favorecido muy positivamente en su

desarrollo y continuas actualizaciones, convirtiéndolo en una de las herramientas más utilizadas por los programadores orientados a Internet. [10]

MySQL se integra perfectamente con Apache como servidor de aplicaciones y PHP como lenguaje de programación Web. La combinación de estos tres elementos es considerada por muchos desarrolladores como la mejor integración para el desarrollo web.

En el caso del CMS Drupal, existe una tendencia extremadamente marcada a desarrollar módulos sobre MySQL, llegando incluso a situaciones donde las funcionalidades de algunos módulos no funcionan con PostgreSQL. A continuación se detallan algunos de ellos:

- Módulo Quiz: Permite generar pruebas online sobre diferentes contenidos.
- Módulo Private Message: Establece la comunicación personal entre usuarios.
- Módulo Backup and Migrate: Construye salvallas del sitio y facilita la migración a otras versiones de Drupal.
- Módulo Gallery: Construye galerías de imágenes con diferentes estilos y formatos.

Tomando en consideración que en las aplicaciones Web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos y que lo que se necesita es un sistema rápido y con pocos usuarios concurrentes, se elige a MySQL como SGBD por reunir las características que más se ajustan al sistema que se quiere implementar.

1.6 Entorno de Desarrollo Integrado (IDE)

Un IDE es un programa que integra varias herramientas que facilitan el desarrollo de software sobre uno o varios lenguajes de programación. Sus características más comunes son: editor de código, herramientas para traceo, consulta a bases de datos, depurador de código y construcción de interfaz gráfica. Entre los IDEs más reconocidos mundialmente se encuentran: Visual Studio .NET, Java para Eclipse y Borland C++. Para desarrollo Web figuran Zend Framework, Zend Studio, Cake PHP, Symfony entre otros. Estos últimos son herramientas poderosas para el desarrollo de software sobre PHP.

Se decidió optar por **Zend Studio para Eclipse 2008** para el desarrollo de este trabajo, debido a que integra lo mejor de las prácticas actuales para desarrollo web, permite la integración con Eclipse y simplifica en gran medida las tareas de integración. Posee herramientas para editar, analizar, optimizar y depurar el código generado. Permite trabajar con bases de datos y otros lenguajes relacionados como *JavaScript*, HTML y XML. Está diseñado para adaptarse a los procesos de las metodologías de desarrollo ágiles.

1.7 Servidor de aplicaciones

En informática se denomina servidor de aplicaciones a un software que proporciona aplicaciones a los equipos o dispositivos cliente, por lo general a través de Internet y utilizando el protocolo *HyperText Transfer Protocol* (HTTP). Se distinguen de los servidores Web por el uso extensivo del contenido dinámico y por su frecuente integración con bases de datos. La ventaja principal de un servidor de aplicaciones es la facilidad para desarrollarlas, puesto que éstas no necesitan ser programadas y en cambio, se arman a partir de módulos provistos por el servidor de aplicaciones. [11]

El servidor Apache inicialmente se basó en el servidor desarrollado en la *National Center for Supercomputing Applications* (NCSA), al que se le añadieron algunos parches. Un parche es un segmento de código que se le adiciona a un programa, con el propósito de arreglar errores o de añadir nuevas funciones. Estos arreglos hicieron de Apache un servidor más competente que el de la NCSA en menos de un año.

Es un software libre que se desarrolla dentro del proyecto *HTTP Server* de la *Apache Software Foundation*, cuya función principal es atender las peticiones de los clientes y enviarse las páginas Web solicitadas. Presenta entre otras características, mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido. Es el servidor más empleado en el mundo, actualmente su porcentaje de utilización supera el 60%. Sus ventajas más significativas son:

- Modular
- Código abierto
- Multiplataforma
- Extensible
- Popular (fácil conseguir ayuda/suporte).[12]

Debido a que PHP se puede utilizar como módulo de Apache, lo que hace que la velocidad de las páginas sea mucho mayor, se decidió utilizar Apache como servidor de aplicaciones.

1.8 Metodologías de desarrollo de software

Una metodología define de forma general como realizar y en qué orden, un grupo de tareas con un objetivo específico. Aplicado al desarrollo de software, divide las tareas en fases o etapas, define las herramientas, técnicas de desarrollo y detalla procesos para llegar a un producto terminado.

Existen dos tipos fundamentales de metodologías de desarrollo de software:

- Las metodologías robustas se basan en la planificación total antes de comenzar a desarrollar el producto y generan una gran cantidad de documentación.
- Las metodologías ágiles por su parte incluyen los cambios en los requerimientos como una parte vital en sus actividades. Prefieren una planificación más descentralizada, generan la documentación estrictamente necesaria para lograr un software funcional y el cliente forma parte esencial del equipo de trabajo.

1.8.1 Metodologías robustas

Una de las metodologías denominadas robustas más conocida y utilizada es **RUP** (*Rational Unified Process*). Integra muchas ventajas de metodologías anteriores a ella, es iterativa e incremental, centrada en la arquitectura y dirigida por casos de uso.

Está diseñada para enfrentar grandes proyectos, con mucho personal y varios equipos de trabajo. Hace énfasis en la documentación y es de mayor uso en sistemas con programación orientada a objetos.

Define cuatro fases y seis flujos de trabajo de trabajo. Las fases son: Inicio, Elaboración, Construcción y Transición, dentro de las cuales se desarrollan de manera iterativa los flujos de trabajo. Cada flujo genera un grupo determinado de artefactos, los cuales son modelados en su mayoría utilizando el lenguaje *Unified Modeling Language* (UML).

RUP comienza modelando el funcionamiento del entorno definiendo las fronteras donde estará el sistema a construir, para lograr una mayor comprensión de los procesos a automatizar.

Una vez terminada la modelación, se capturan los requerimientos funcionales y no funcionales. Los funcionales son convertidos en casos de uso y estos se agrupan según diversos criterios. Cada uno de los casos de uso definidos se detalla al máximo mediante artefactos que constituyen la información de entrada para el desarrollo del sistema.

Cada funcionalidad desarrollada es probada con pruebas de caja blanca y pruebas caja negra. Las pruebas de caja blanca verifican la calidad del código fuente, mientras que las pruebas de cada negra comprueban el funcionamiento del sistema.

1.8.2 Metodologías ágiles

En los últimos años, las metodologías ágiles que más se han popularizado y que cuentan con resultados probados son Scrum y Extreme Programming.

Scrum: Se basa en la planificación de tareas modulares y pequeñas, que son realizadas por equipos de desarrollo, en períodos de trabajo intensivo llamados *Sprint*. Esta metodología define tres conceptos fundamentales: herramientas, prácticas, roles y responsabilidades.

Las **herramientas** contienen las tareas a entregar en cada momento del desarrollo del software. Estos listados están sujetos a cambios frecuentes.

Las **prácticas** se basan en mantener un desarrollo ágil y que se ajuste a los cambios.

Los **roles y responsabilidades** controlan el cumplimiento de las tareas, retroalimentándose de los resultados.

Esta metodología de desarrollo consta de tres fases:

- *.Pregame:* Se capturan los requisitos fundamentales, se crea el *Product Backlog List* y se prepara el primer *sprint (planning)*, en el cual se define una propuesta de arquitectura, diseño y estándares de desarrollo.
- *Development:* Es la etapa de desarrollo intensivo, donde ocurren los cambios impredecibles. Scrum posee una serie de prácticas para controlar las variables de tiempo, calidad, tecnología, recursos, requerimientos y herramientas para adaptarse a dichos cambios de forma flexible.
- *PostGame:* Se llega a un consenso sobre la terminación del producto. Se realizan las pruebas, la integración y la documentación.

Los roles fundamentales de SCRUM son:

- *SCRUM Master:* Vela porque el desarrollo del producto se realice con el buen uso de las prácticas de Scrum. Su principal tarea es eliminar impedimentos y reducir riesgos. Este rol es desempeñado generalmente por el líder del proyecto.
- *Product Owner:* Vela por la integridad del *Product Backlog List* y se asegura de que las tareas a desarrollar tributen a la visión del proyecto.
- *SCRUM Team:* Es la base del desarrollo Scrum. Es un grupo de 5 a 10 personas, auto-dirigido y que realiza estimación de esfuerzos, *sprint Backlog* y señala obstáculos a ser removidos durante el proceso de desarrollo. Está compuesto por programadores, arquitectos, y diseñadores.
- *Customer:* Es la representación del cliente que trabajará junto al equipo de desarrollo.
- *Management:* Es el responsable de tomar las decisiones finales en cuanto a convenciones y estándares. Trabaja muy unido al *SCRUM Master*.

El ciclo de vida de Scrum es iterativo y se desarrolla en las manos de equipos auto-dirigidos. No define prácticas de ingeniería específica, delimita pocos artefactos y flujos de trabajo fundamentales. Cada iteración se denomina *Sprint*.

Product Backlog List. Es la lista de requerimientos, funcionalidades o historias de usuario. Se elabora según se vaya desarrollando el sistema, lo que significa que mientras exista el producto existirá el *Product Backlog*. Las historias, a medida que van surgiendo, se agregan a esta lista y son implementadas en orden según su importancia o necesidad. Generalmente representan requerimientos funcionales, pero en ocasiones también representan requerimientos no funcionales que no están relacionados directamente con ninguna historia de usuario específica, pero aun así son necesarias para la entrega final del producto. Este tipo de historias se denominan “historias técnicas”.

Para calcular la estimación inicial, o lo que es lo mismo, cuánto trabajo es necesario para implementar una historia, la metodología plantea que se realice una valoración teniendo en cuenta que: “si tuvieran el número óptimo de personas para esta historia (ni muchos ni pocos, típicamente 2) y se encierran en una habitación con cantidad de comida, y trabajan sin distracciones, ¿en cuántos días saldrían con una implementación terminada, demostrable, testeada y liberable?”. Por ejemplo: si la respuesta es “con 3 personas encerradas en una habitación llevaría 4 días”, entonces la estimación inicial es 12 puntos. La unidad son “puntos de historia” y usualmente corresponde a “días-persona ideales”. [13]

Sprint. Es un período de tiempo que oscila entre una y seis semanas, donde se implementan las funcionalidades y se realizan las tareas definidas para el *sprint*. En esta fase, el producto experimenta los cambios señalados por el cliente en el diseño, codificación y pruebas. Además, la arquitectura es propensa a variar en dependencia de dichos cambios.

Cada *sprint* es diseñado con mucho detalle, en una reunión de inicio de aproximadamente 4 horas de duración y es evaluado diariamente con el equipo en reuniones de 15 minutos de duración.

En estas reuniones se evalúa el desempeño del día anterior y se planifican las tareas a desarrollar en el presente día con las siguientes preguntas:

- ¿Qué hiciste ayer?
- ¿Qué harás hoy?
- ¿Qué obstáculos ves en tu camino?

Es importante aclarar que mientras el *sprint* dure, no pueden introducirse cambios de ningún tipo. Cualquier requerimiento nuevo que surja, se almacena en el *Product Backlog List* hasta que finalice el *sprint*.

Sprint Backlog: Es la lista de tareas a realizar durante el *sprint*. Su contenido debe ser cuidadosamente analizado y planificado, para reducir al máximo los desvíos durante el *sprint*. Es posible aumentar o disminuir la cantidad de elementos de este listado mientras no afecte a los requerimientos que serán implementados en esa iteración.

Burndown Chart: Es un diagrama que muestra el cumplimiento de las tareas a realizar en el *sprint*. Su gráfica se construye diariamente, dependiendo de los resultados arrojados por las reuniones diarias. El valor de las tareas está dado en puntos de historias de usuarios y su cumplimiento se expresa por la cantidad de puntos desarrollados en un día. (Ver anexos 2 y 3)

Stabilization sprints: Son *sprints* que se concentran en la corrección de defectos. Generalmente se ejecutan cuando el producto está por liberarse. [14]

XP: Es una metodología centrada en el desarrollo de las relaciones inter-personales y en una retroalimentación constante con el cliente. Posee tres apartados fundamentales: Historias del usuario, que contiene los requerimientos del software según el cliente, roles y por último, procesos y prácticas. Los procesos están definidos por seis fases que cubren desde la exploración hasta la muerte del proyecto. Las prácticas se caracterizan por ser objetivas y muy específicas, logrando que se auto complementen en la práctica.

Historias de usuario: Son generalmente notas en hojas de papel que expresan requerimientos funcionales y no funcionales del software. Cada una de las historias de usuario puede desecharse o sustituirse por otra más específica. Su principal característica es que son lo suficientemente delimitadas, para que un equipo la desarrolle en unas pocas semanas. Estas se dividen en tareas de programación y son asignadas para ser realizadas en una iteración.

Los principales roles de XP son los siguientes:

- Programador: Desarrolla el código del sistema.
- Cliente: Es la fuente de las historias de usuarios. Determina cuales son las pruebas fundamentales para garantizar un buen funcionamiento del producto.

- Encargado de pruebas: Trabaja junto al cliente en la elaboración de las pruebas funcionales y las realiza regularmente.
- Entrenador: Es el experto del equipo. Se encarga de guiar al resto en un correcto proceso de desarrollo.
- Consultor: Miembro externo con conocimientos avanzados de algún tema necesario en el proyecto.
- Gestor: Coordina las actividades del equipo.

El ciclo de vida de XP consta de seis fases:

- Fase I (Exploración): El equipo se familiariza con el proyecto y las tecnologías necesarias. Al mismo tiempo, el cliente desarrolla la mayor cantidad posible de historias de usuarios. La duración de esta fase depende del nivel de conocimientos del equipo sobre las tecnologías a utilizar.
- Fase II (Planificación de entrega): El cliente prioriza las historias de usuarios y el equipo hace una estimación de duración para las mismas. Cada entrega debe realizarse en unos pocos meses.
- Fase III (Iteraciones): Cada iteración incluye historias de usuarios que no habían sido incluidas en la iteración anterior, corrige defectos encontrados y asume tareas anteriormente sin terminar.
- Fase IV (Producción): Al sistema se le realiza una serie de pruebas más rigurosas para ser trasladado al entorno del cliente.
- Fase V (Mantenimiento): Cada última versión se mantiene funcionando mientras el equipo desarrolla la siguiente. Esto requiere realizar tareas de soporte al cliente.
- Fase VI (Muerte del proyecto): Cuando el cliente no tiene más historias de usuarios que agregar y se comienza a refinar el rendimiento y otros aspectos de interés. Otra causa puede ser la cancelación del proyecto.

Algunas prácticas de desarrollo de XP son:

- Entregas pequeñas: Se desarrollan versiones incompletas del producto, pero con funcionalidades totalmente operativas que añaden valor al mismo.
- Diseño simple: Reduce al mínimo posible el código y la cantidad de funciones. Además trata de no duplicar lógica de negocio alguna.

- Refactorización: Consiste en la revisión constante de código para reestructurarlo y hacerlo más entendible, eliminar duplicación de algoritmos y funciones.
- Programación en parejas: Las principales ventajas de esta práctica son el poco tamaño del código, menor tasa de errores, solución rápida de los problemas y mejor diseño.
- Propiedad colectiva del código: Cada programador puede cambiar cualquier fragmento de código en cualquier momento. Esto estimula la creatividad del equipo.
- 40 horas a la semana: No trabajar horas extras de manera continua para no disminuir la productividad del equipo. [15]

1.8.3 ¿Por qué Scrum y XP?

Como metodología de desarrollo se resolvió utilizar la combinación de dos metodologías ágiles: Scrum y *Extreme Programming* (XP). Debido a que el desarrollo del producto final debe realizarse en un breve período de tiempo y la herramienta seleccionada es un CMS muy configurable, esta combinación se ajusta perfectamente a las necesidades y condiciones del equipo de desarrollo.

XP define la dirección de los equipos de manera muy básica y se centra casi en su totalidad en las tareas de desarrollo, preocupándose por hacer mejoras continuas al producto. Sin embargo, no cuenta con ningún rol para la documentación, ni un proceso claro para la gestión de información. Para un proyecto de las características propuestas, es necesario un cierto nivel de documentación que XP no está preparado para proporcionar. Scrum es superior en este aspecto, por lo que puede complementar perfectamente esta necesidad.

La combinación de SCRUM y XP se traduce en un proceso de desarrollo del software marcado por continuas entregas al cliente, el cual es consciente del avance del proyecto. La documentación generada es la estrictamente necesaria, por lo que es recomendable para proyectos pequeños o medianos que requieran una entrega inmediata y que cuentan con pocos integrantes en su equipo de trabajo.

Por su parte, RUP es una metodología diseñada para proyectos complejos, orientados a objetos y por la gran cantidad de documentación generada en cada flujo de trabajo, unido a que define un buen número de roles, lo hace poco factible para equipos de menos de 15 personas con plazos de entrega cortos.

1.8.4 Unified Modeling Language (UML)

UML es un lenguaje de representación visual, que se utiliza para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. Su objetivo principal es

visualizar, especificar, construir y documentar, los artefactos de un sistema que involucra una gran cantidad de software y es con este fin que se utiliza en el trabajo.

1.8.5 Visual Paradigm

Como herramienta para modelar el sistema se escogió Visual Paradigm, debido a que es multiplataforma, contiene una interfaz gráfica muy intuitiva y de fácil aprendizaje, que apoya las últimas notaciones de UML. Permite la modelación de sistemas simples y avanzados, además de funcionalidades añadidas como ingeniería inversa y otras. Es una herramienta que resiste el ciclo de vida completo del desarrollo de software, desde análisis y diseño hasta despliegue.

1.9 Conclusiones

En la actualidad, existen en el mundo un número considerable de sistemas que automatizan gran parte de la gestión de incidencias. Estos sistemas tributan generalmente a órganos relacionados con la ley y el orden como policía, fuerzas armadas y fiscalía. Sin embargo, dicha gestión en la UCI tiene muchos matices únicos que impiden la utilización de otros sistemas ya implementados, por lo que se determinó realizar un sistema que se ajuste a las necesidades específicas del centro.

Teniendo en cuenta que el producto final debe ser ajustable a futuros cambios, se optó por la combinación de SCRUM y XP como metodología de desarrollo, pues estas combinan un desarrollo ágil del producto con un excelente control de cambios. Entre las herramientas que se escogieron para el desarrollo del sitio Web se encuentra Drupal, por ser un sistema gestor de contenidos libre que cumple con los estándares internacionales de desarrollo Web. El servidor Apache está ubicado a la vanguardia en aplicaciones desarrolladas con software libre, lo que unido a la utilización de PHP y MySQL, garantizan una codificación robusta y rápida.

CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

2.1 Introducción

En este capítulo se abordarán los aspectos fundamentales de la propuesta de solución de este trabajo. Se especifican los requisitos que debe cumplir la aplicación en el *Product Backlog*. Se describen las características del sistema en cuanto a la arquitectura y los patrones de diseño utilizados y algunos parámetros relacionados con la seguridad, el tratamiento de excepciones y los estándares de codificación utilizados en su construcción.

2.2 Descripción de los procesos de gestión de incidencias

Este trabajo pretende modelar los procesos de gestión de incidencias que se llevan a cabo en la Oficina de Seguridad y Protección de la UCI. Para ello es necesario conocer la forma en que se realizan en la actualidad:

Cuando una persona detecta una incidencia debe informarla inmediatamente, preferiblemente a cualquier Puesto de Mando de la UCI (MININT, SEPCOM, SP). También pueden dirigirse a la guardia obrera o estudiantil, o al SEPCOM que más cerca se encuentre del lugar donde ocurrieron los hechos y estos se encargaran de informarlo al Puesto de Mando UCI. Después de reportarse el incidente, se procede a su esclarecimiento.

- Los datos concernientes a las incidencias (lugar del hecho, infractores, persona que lo detecta, entre otros) se recogen inmediatamente sean reportadas al Puesto de Mando UCI. Después de informado el suceso, se clasifica según su importancia. Seguidamente a su clasificación, si la incidencia tiene una alta importancia, se envía por correo toda la información concerniente a la misma al Especialista de Seguridad y Protección, que es el encargado de realizar la investigación en la que se deben aclarar los hechos. De lo contrario los datos son archivados en el puesto de mando UCI, y al final del día se le envía todos los datos de los incidentes ocurridos clasificados de menor importancia.
- Después de recibida la información, el Especialista de Seguridad y Protección elabora el expediente. Para esto imprime una planilla de expediente que tiene elaborada, en la que escribe los datos que le fueron enviados y más tarde procede a realizar la investigación. Según

efectúa sus averiguaciones incluye en el expediente los datos que reúne. Después de aclarada la incidencia el expediente se archiva.

- Para saber si una determinada persona a incurrido en una incidencia, se procede a realizar una búsqueda manual en los documentos archivados. Si el expediente se archivó recientemente, puede que el Especialista de Seguridad y Protección recuerde la ubicación exacta del mismo. De no ser así, esta actividad requerirá mucho tiempo debido a la cantidad de expedientes almacenados. Esta forma de búsqueda incluye un alto margen de error.
- Para elaborar el informe mensual se realiza una búsqueda manual, en la que se seleccionan las incidencias que ocurrieron en el mes del que se necesita la información. Después de obtener los datos necesarios se procede a realizar los cálculos manualmente. Se elabora el informe introduciendo los datos necesarios para el mismo.
- Si se desea obtener un reporte semanal, se realiza una búsqueda manual de las incidencias que ocurrieron en esa semana y se clasifican según robo, apropiación, violación, entre otros. Posteriormente se realizan los cálculos y se introducen los datos en la planilla de reporte. La elaboración de este reporte es relativamente más sencilla que el Informe Mensual, por necesitar una menor cantidad de información para elaborarlo.

Como se puede apreciar en la descripción de los procesos existen varios problemas que impiden que estos se desarrollen de forma rápida y eficientemente.

2.3 Propósito del sistema

En este momento en la universidad no existe ningún sistema que gestione los datos relacionados con los incidentes ocurridos. Es por eso que existen grandes dificultades en la realización de los procesos anteriormente descritos:

- Dificultad en el trabajo con los expedientes, ya que estos no son almacenados hasta aclarar el suceso.
- Existe una gran cantidad de archivos físicos, los cuales ocupan gran espacio en el local, además de que pueden deteriorarse con el tiempo. Esto provoca la disminución de la credibilidad de los datos.
- Dificultad en la realización de búsquedas para conocer si una determinada persona ha incurrido en una incidencia o para obtener datos específicos de los incidentes, debido a que ningún

procesos esta automatizado. Esta información se necesita para la confección de los reportes semanales y los informes mensuales.

- Por la existencia de gran cantidad de archivos que se manipulan a diario, es imposible lograr la organización y rapidez en el trabajo.
- Los cálculos necesarios para la confección de los reportes e informes se realizan manualmente, esto da cabida a que exista un margen de error en los mismos.

Sobre la base de eliminar esas dificultades y debido a la necesidad que tiene la universidad de gestionar de manera rápida y eficiente la información de las incidencias ocurridas, se desarrollará el Sistema de Gestión de Incidencias para la UCI. El sistema tendrá como objetivo automatizar los procesos de gestión de incidencias, con lo que se pretende eliminar parte de los problemas presentes en los mismos y agilizar su desarrollo.

2.4 Descripción del sistema propuesto

El sistema propuesto centralizará su función en la creación de expedientes por incidencias que se detecten, en la obtención de cualquier información que se necesite, en la búsqueda de personas para saber si ha cometido alguna incidencia, en el trabajo con los expedientes, es decir, el seguimiento de ellos hasta que el suceso sea esclarecido y en la creación de reportes e informes.

2.4.1 Roles de usuarios

A continuación se relacionan los roles planeados para actuar sobre el Sistema de Gestión de Incidencias:

Tabla 2.1 Definición de los roles de usuarios.

Roles	Justificación
Especialista	Se encarga de crear los expedientes a partir de las incidencias, de registrar los infractores y de crear los diferentes reportes e informes. Es el único autorizado a eliminar información del sistema. Define los usuarios y direcciones IP que podrán acceder al sistema. El rol Especialista representa al Especialista de Seguridad y Protección.

Técnico	Es el encargado de reportar incidencias y además puede buscarlas. El rol representa a los Oficiales Operativos.
Administrador	Tiene acceso a todas las funcionalidades del CMS con el fin de construir la aplicación. Al concluir la implementación del sistema termina sus actividades, debido a que ningún usuario final tendrá este rol.

2.4.2 Arquitectura y diseño

Uno de los objetivos que se persigue con la implementación de un sistema para gestionar incidencias, es permitir el acceso al mismo por parte de los clientes desde cualquier computador, sin necesidad de que la aplicación se instale en las computadoras destinadas para realizar el trabajo. El desarrollo de este tipo de aplicaciones es un tanto difícil, por eso es necesario realizar un correcto diseño que permita darle una arquitectura apropiada al sistema, con el fin de facilitar la construcción del mismo.

En múltiples foros y espacios de discusión especializada, muchos desarrolladores asignan a Drupal una arquitectura modular. Sin embargo, a pesar de lo sugerente y clarificador de esta clasificación, la misma no se encuentra definida o avalada por ningún estándar internacional. A continuación se describen los estilos arquitectónicos más utilizados:

- Flujo de datos: Se centran en la transformación de los datos de entrada.
 - Sistemas de filtros y tuberías.
 - Procesamiento por lotes.
- Llamada y retorno: Reflejan la estructura del lenguaje de programación.
 - Programación principal y subrutina.
 - Sistemas orientados a objetos.
 - Organización por capas.
- Centrados en datos: Se caracterizan por acceder de manera compartida a una fuente central de datos.
 - Repositorios.
 - Pizarras.

- Máquinas virtuales: Simulan funcionalidades no nativas a un entorno determinado.
 - Sistemas basados en reglas.
- Componentes independientes: Los componentes del sistema funcionan de una forma autónoma en menor o mayor medida y se comunican mediante mensajes.
 - Cliente/servidor.
 - Sistemas orientados a eventos.
 - Sistemas orientados a servicios.

A simple vista destacan dos categorías de estilos arquitectónicos capaces de ajustarse a una aplicación construida utilizando Drupal. Estas son: Llamada y retorno y Componentes independientes. Antes de analizarlas con más profundidad, es preciso describir más detalladamente el funcionamiento de Drupal.

En Drupal existe una clasificación básica llamada nodo, que contiene cualquier información almacenada. Cada nodo en específico pertenece a un tipo de contenido determinado. Los tipos de contenido son plantillas que definen como almacenar la información de los nodos. A manera de ejemplo, en Drupal, el tipo de contenido *Page* (Página) guarda un nodo de información que contiene un título y un cuerpo del contenido. Por otra parte, un tipo de contenido *Persona*, podría guardar nodos de información con los campos nombre, apellidos, carnet de identidad, entre otros.

Las funcionalidades de Drupal están separadas en módulos. Cada módulo gestiona su generación de interfaz, lógica de negocio y acceso a datos. Estos módulos pueden clasificarse de manera intuitiva en dos categorías: los que componen el *Core* y los que extienden las funcionalidades de la aplicación.

En el *Core* se encuentra lo necesario para que el sistema comprenda el trabajo de los demás módulos y el CMS funcione. El resto de los módulos son desarrollados por cualquier miembro de la comunidad de Drupal.

Para que el *Core* interprete correctamente el funcionamiento de cada módulo, estos presentan un grupo de funciones llamadas *hooks* (ganchos). Cada *hook* tiene una función específica, como definir el menú de opciones, el menú de administración, la generación de las páginas al usuario, entre otras. Tienen parámetros muy bien definidos. Cuando el sistema necesita mostrar cualquier información al usuario, llama a todos los *hooks* que tienen el nombre adecuado y luego define cuál de ellos se encuentra en el módulo correcto y lo ejecuta.

Una vez explicado de forma elemental el funcionamiento de Drupal, se puede concluir que no es un sistema orientado a objetos, aunque la estructura de nodos y tipos de contenido sugiera un

comportamiento similar. Tampoco puede ubicarse en una aplicación construida por capas, pues como se dijo anteriormente, cada módulo gestiona su propia interfaz, lógica de negocio y acceso a datos. No existe una “capa” uniforme a ningún nivel de la aplicación.

Sin embargo, un sistema desarrollado en Drupal se ajusta muy bien a la definición de Componentes independientes, dado en gran medida por el nivel de independencia de sus módulos. Los hooks tienen el mismo nombre en cada uno de los módulos, pero describen un comportamiento diferente, ideal para ubicarse como un sistema orientado a eventos.

Drupal presenta comportamientos descritos en varios patrones de diseño:

- *Singleton*: Limita a solo una la cantidad de instancias de una clase y proporciona un punto de acceso global a la misma. Los nodos de Drupal son tratados como objetos, de los cuales solo existe una única instancia que se referencia teniendo en cuenta los niveles de acceso permitidos por el usuario.
- *Observer*: Plantea las relaciones entre objetos de forma que cuando alguno de ellos varía alguna de sus propiedades, el observador informa al resto de los objetos del cambio. Es una forma eficiente de capturar eventos, al igual que el tratamiento que da Drupal a sus *hooks*.
- *Command*: Permite realizar peticiones a un objeto sin conocer la petición en sí, ni conocer el destino de la misma. Se logra encapsulando los datos de la petición en un objeto. Es útil cuando los parámetros de la petición pueden ser de un amplio espectro. Drupal no tiene objetos de clases pero utiliza sus tablas en la base de datos para guardar algunas de las acciones de los usuarios.
- *Reflection Pattern*. Este patrón es predominante en Drupal. Permite adaptarse dinámicamente a los cambios en las solicitudes a la información almacenada. Estos cambios pueden afectar la estructura de la información y otros aspectos importantes, por lo que el uso de este patrón permite a una aplicación mantenerse confiable y segura en un entorno altamente variable.

2.4.3 Despliegue

El modelo de despliegue define la arquitectura física del sistema por medio de nodos interconectados. Estos nodos son elementos hardware sobre los cuales pueden ejecutarse los elementos software. La figura 1 presenta el Modelo de Despliegue propuesto teniendo en cuenta los recursos con los que contaba el cliente:

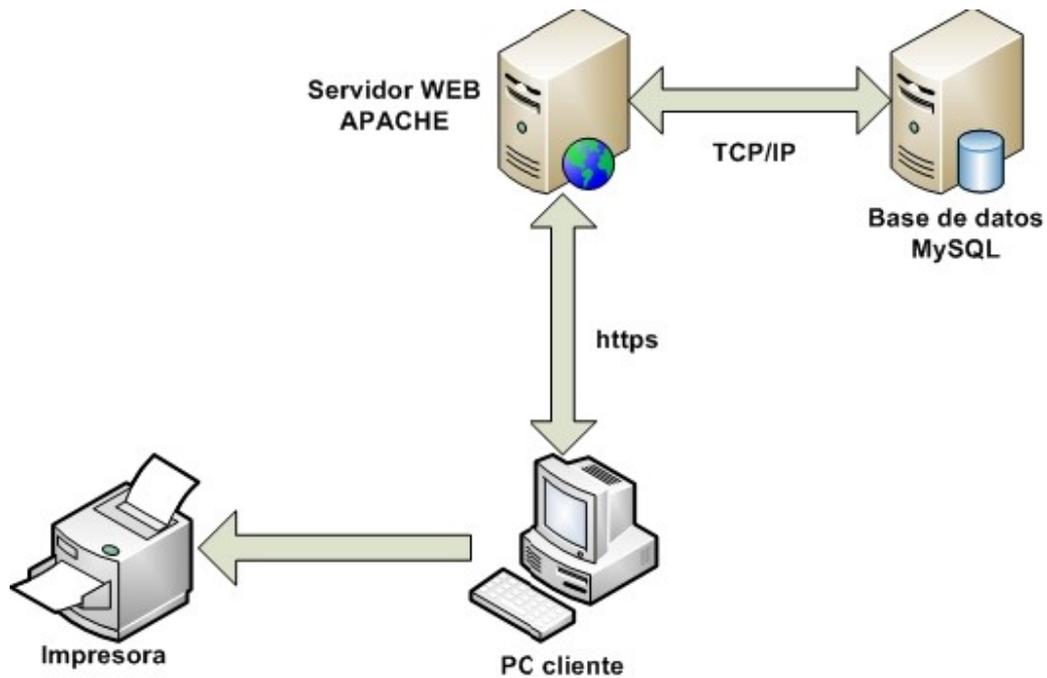


Figura 1. Diagrama de despliegue.

2.4.4 Seguridad del Sistema

Al tratarse de un sistema Web, las mayores amenazas a la seguridad provienen de los datos que se introducen en los formularios del mismo. Esto se refiere principalmente a *scripts* que traspasan los permisos asignados a los usuarios finales y a la inyección de código SQL. El Core de Drupal provee validación de los datos desde el lado del servidor, controlando estas agresiones.

La barra de direcciones es también un blanco frecuente de ataques. Estos consisten en escribir las direcciones a las que se desea acceder en la misma barra de direcciones y burlar los permisos asignados por el administrador. Para evitar esto, Drupal cuenta con el módulo *phpid*, que regula los permisos de las *URLs* solicitadas al servidor.

La tercera línea defensiva contra ataques malintencionados constituye el trabajo y permisos por roles. Cada usuario final del sistema accede al mismo para realizar tareas específicas de su rol. Una estricta asignación de permisos permite controlar el acceso de cada usuario solamente a las áreas que le están asignadas. Por último, el sistema regula las direcciones IP, los usuarios y las direcciones de correo electrónico que pueden acceder por la red. Estas reglas de acceso pueden ser fácilmente añadidas, modificadas y eliminadas por el Especialista en Seguridad, el cual deviene en administrador del sistema.

Esta medida posibilita habilitar el acceso de una nueva computadora o usuario. Por supuesto, también permite lo opuesto, la negación del servicio a usuarios, ordenadores y direcciones de correo electrónico.

2.4.5 Product Backlog

En el *Product Backlog* se enuncian y describen todos los requisitos o funcionalidades que deben estar presentes en la aplicación. Se llaman historias de usuario o elementos de la pila y se describen utilizando los términos del cliente. El *Product Backlog* se confecciona con el fin de identificar las necesidades del cliente para que se pueda desarrollar un producto que cumpla con sus expectativas.

Tabla 2.2 Historia de usuario Autenticar Usuarios.

Historia de Usuario	
Número:1	Usuario: Especialista de Seguridad y Protección – Oficial Operativo
Nombre historia: Autenticar Usuarios	
Estimación inicial: 4	Iteración asignada: 4
Descripción: Para entrar al sistema, el mismo debe solicitar el nombre de usuario y contraseña.	
Cómo probarlo: Crear un rol y asignarlo a un usuario. Comprobar que pueda acceder al sistema.	
Notas:	
<ul style="list-style-type: none"> - Los usuarios solo podrán autenticarse por el dominio de la universidad. - Se necesitan los datos del LDAP_ UCI. 	

Tabla 2.3 Historia de usuario Administrar Usuarios.

Historia de Usuario	
Número: 2	Usuario: Especialista de Seguridad y Protección
Nombre historia: Administrar Usuarios	
Estimación inicial: 4	Iteración asignada: 4
Descripción: El sistema debe permitir asignarle permisos a los roles del sistema y los roles a los usuarios para controlar que la información sea accedida por las personas autorizadas.	
Cómo probarlo: A un usuario registrado en la base de datos asignarle un rol. Cuando el usuario vuelva a autenticarse comprobar que cuente con los permisos del rol asignado.	
Notas:	
<ul style="list-style-type: none"> - Cada usuario del sistema tendrá acceso según su rol. 	

Tabla 2.4 Historia de usuario Buscar Contenido.

Historia de Usuario	
Número: 3	Usuario: Especialista de Seguridad y Protección
Nombre historia: Buscar Contenido	
Estimación inicial: 40	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de buscar los diferentes tipos de contenido: incidencia, expediente, infractor, reporte semanal e informe mensual.	
Cómo probarlo: Seleccionar el tipo de contenido que desea buscar, filtrar los campos por los que desea realizar la búsqueda y comprobar el resultado.	
Notas:	
<ul style="list-style-type: none"> - Los campos para filtrar deben permitir diferentes operadores de comparación, como igual, contiene y no es igual. 	

Tabla 2.5 Historia de usuario Mostrar Contenido.

Historia de Usuario	
Número: 4	Usuario: Especialista de Seguridad y Protección
Nombre historia: Mostrar Contenido	
Estimación inicial: 40	Iteración asignada: 3
Descripción: El sistema debe generar los ficheros PDF correspondientes a cada tipo de contenido: incidencia, expediente, infractor, reporte semanal e informe mensual.	
Cómo probarlo: Acceder a los elementos mencionados en la descripción y comprobar que se genera el archivo PDF con los datos correctos.	
Notas:	

Tabla 2.6 Historia de usuario Reportar una Incidencia.

Historia de Usuario	
Número: 5	Usuario: Especialista de Seguridad y Protección – Oficial Operativo
Nombre historia: Reportar una Incidencia	
Estimación inicial: 8	Iteración asignada: 1
Descripción: El sistema debe permitir la creación de nuevas incidencia, introduciendo manualmente todos los datos necesarios para su procesamiento: asunto, quién recibe el reporte, fecha en que se detectó la incidencia, descripción, fecha y hora en que se informó la incidencia y el nombre, apellidos, carnet de identidad y solapín, del detector y del informante.	

Cómo probarlo: Llenar y enviar los datos que se solicitan para reportar la incidencia. Comprobar que los datos fueron guardados.
Notas:

Tabla 2.7 Historia de usuario Modificar una Incidencia.

Historia de Usuario	
Número: 6	Usuario: Especialista de Seguridad y Protección
Nombre historia: Modificar una Incidencia	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de modificar los datos de una incidencia, después de haber buscado y localizado la incidencia que desea modificar.	
Cómo probarlo: Acceder al formulario que contiene los datos de la incidencia cuando fue reportada. Comprobar que se muestran correctamente. Cambiar la tercera parte de los valores mostrados y enviarlos. Comprobar que los datos fueron actualizados.	
Notas:	

Tabla 2.8 Historia de usuario Eliminar una Incidencia.

Historia de Usuario	
Número: 7	Usuario: Especialista de Seguridad y Protección
Nombre historia: Eliminar una Incidencia	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de eliminar una incidencia, después de haber buscado y localizado la incidencia que desea eliminar.	
Cómo probarlo: Seleccionar y eliminar una incidencia. Comprobar que no quede registro de la misma.	
Notas:	

Tabla 2.9 Historia de usuario Crear un Expediente.

Historia de Usuario	
Número: 8	Usuario: Especialista de Seguridad y Protección
Nombre historia: Crear un Expediente	
Estimación inicial: 8	Iteración asignada: 1

<p>Descripción: El sistema debe permitir la creación de un expediente a partir de una incidencia registrada anteriormente, introduciendo manualmente todos los datos necesarios para su procesamiento: elaborado por, tipificación (tipo de incidencia), número del expediente, fecha y hora de elaboración, estado (estado del expediente), lugar del hecho, descripción, medios afectados, protección del lugar, esclarecimiento (nivel de solución dado a la incidencia) y clasificación. Debe permitir la búsqueda de incidencias para crearles expedientes.</p>
<p>Cómo probarlo: Seleccionar una incidencia. Acceder a la opción "Crear expediente". Llenar y enviar todos los datos del formulario. Comprobar si los datos fueron guardados.</p>
<p>Notas:</p> <ul style="list-style-type: none"> - Solo se creará un nuevo expediente a las incidencias que no tengan expediente creado.

Tabla 2.10 Historia de usuario Modificar un Expediente.

Historia de Usuario	
Número: 9	Usuario: Especialista de Seguridad y Protección
Nombre historia: Modificar un Expediente	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de modificar los datos de un expediente, después de haber buscado y localizado el expediente que desea modificar.	
Cómo probarlo: Acceder al formulario que contiene los datos del expediente seleccionado y modificar la tercera parte de los datos. Enviar y comprobar que se guardaron las modificaciones correctamente.	
Notas:	

Tabla 2.11 Historia de usuario Eliminar un Expediente.

Historia de Usuario	
Número: 10	Usuario: Especialista de Seguridad y Protección
Nombre historia: Eliminar un Expediente	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de eliminar un expediente, después de haber buscado y localizado el expediente que desea eliminar.	
Cómo probarlo: Seleccionar y eliminar un expediente. Verificar que no queden registros en la base de datos.	
Notas:	

Tabla 2.12 Historia de usuario Adicionar un Infractor.

Historia de Usuario	
Número: 11	Usuario: Especialista de Seguridad y Protección
Nombre historia: Adicionar un Infractor	
Estimación inicial: 8	Iteración asignada: 1
Descripción: El sistema debe brindar la posibilidad de adicionar un nuevo infractor a un expediente, introduciendo manualmente todos los datos necesarios para su procesamiento: nombre(s) y apellidos, solapín, ocupación laboral, carnet de identidad y procedencia.	
Cómo probarlo: Acceder al formulario de creación de un infractor. Llenar y enviar los datos. Comprobar que los datos fueron guardados correctamente.	
Notas: Si el infractor está registrado en el sistema, basta con declarar el nombre del mismo.	

Tabla 2.13 Historia de usuario Modificar un Infractor.

Historia de Usuario	
Número: 12	Usuario: Especialista de Seguridad y Protección
Nombre historia: Modificar un Infractor	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de modificar los datos del historial de un infractor, después de haber buscado y localizado el infractor cuyos datos desea modificar.	
Cómo probarlo: Acceder al formulario que contiene los datos del infractor seleccionado y modificar la tercera parte de los datos. Enviar y comprobar que se guardaron las modificaciones correctamente.	
Notas:	

Tabla 2.14 Historia de usuario Eliminar un Infractor.

Historia de Usuario	
Número: 13	Usuario: Especialista de Seguridad y Protección
Nombre historia: Eliminar un Infractor	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de eliminar el historial de un infractor, después de haber buscado y localizado el infractor cuyo historial desea eliminar.	
Cómo probarlo: Seleccionar y eliminar el historial de un infractor. Verificar que no queden registros en la base de datos.	
Notas:	

Tabla 2.15 Historia de usuario Crear un Reporte Semanal.

Historia de Usuario	
Número: 14	Usuario: Especialista de Seguridad y Protección
Nombre historia: Crear un Reporte Semanal	
Estimación inicial: 8	Iteración asignada: 1
Descripción: El sistema debe permitir la creación de un reporte semanal a partir de los expedientes de las incidencias reportadas en esa semana, introduciendo manualmente todos los datos necesarios para su procesamiento: semana, encabezado, nombre del año en curso y la fecha de inicio y final de la semana que se va a analizar.	
Cómo probarlo: Acceder al formulario de creación de un reporte semanal. Llenar y enviar los datos. Comprobar que los datos se guardaron correctamente.	
Notas:	

Tabla 2.16 Historia de usuario Modificar un Reporte Semanal.

Historia de Usuario	
Número: 15	Usuario: Especialista de Seguridad y Protección
Nombre historia: Modificar un Reporte Semanal	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de modificar los datos de un reporte semanal, después de haber buscado y localizado el reporte cuyos datos desea modificar.	
Cómo probarlo: Acceder al formulario que contiene los datos de reporte seleccionado y modificar la tercera parte de los valores. Enviar y comprobar que las modificaciones se guardaron correctamente.	
Notas:	

Tabla 2.17 Historia de usuario Eliminar un Reporte Semanal.

Historia de Usuario	
Número: 16	Usuario: Especialista de Seguridad y Protección
Nombre historia: Eliminar un Reporte Semanal	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de eliminar un reporte semanal, después de haber buscado y localizado el reporte que desea eliminar.	

Cómo probarlo: Seleccionar y eliminar un reporte. Comprobar que no queden registros en la base de datos.
Notas:

Tabla 2.18 Historia de usuario Crear un Informe Mensual.

Historia de Usuario	
Número: 17	Usuario: Especialista de Seguridad y Protección
Nombre historia: Crear un Informe Mensual	
Estimación inicial: 8	Iteración asignada: 1
Descripción: El sistema debe permitir la creación de un informe mensual a partir de los expedientes de las incidencias reportadas en ese mes, introduciendo manualmente todos los datos necesarios para su procesamiento: encabezado del informe, nombre del año en curso, el mes que se va analizar y la fecha inicial y final del mismo.	
Cómo probarlo: Acceder al formulario de creación de un informe mensual. Llenar y enviar los datos. Comprobar que los datos se guardaron correctamente.	
Notas:	

Tabla 2.19 Historia de usuario Modificar un Informe Mensual.

Historia de Usuario	
Número: 18	Usuario: Especialista de Seguridad y Protección
Nombre historia: Modificar un Informe Mensual	
Estimación inicial: 2	Iteración asignada: 2
Descripción: El sistema debe brindar la posibilidad de modificar los datos de un informe mensual, después de haber buscado y localizado el informe cuyos datos desea modificar.	
Cómo probarlo: Acceder al formulario que contiene los datos de informe seleccionado y modificar la tercera parte de los valores. Enviar y comprobar que las modificaciones se guardaron correctamente.	
Notas:	

Tabla 2.20 Historia de usuario Eliminar un Informe Mensual.

Historia de Usuario	
Número: 19	Usuario: Especialista de Seguridad y Protección
Nombre historia: Eliminar un Informe Mensual	
Estimación inicial: 2	Iteración asignada: 2

Descripción: El sistema debe brindar la posibilidad de eliminar un informe mensual, después de haber buscado y localizado el informe que desea eliminar.
Cómo probarlo: Seleccionar y eliminar un informe. Comprobar que no queden registros en la base de datos.
Notas:

Tabla 2.21 Historia técnica Diseñar interfaz.

Historia de Usuario	
Número: 20	Usuario: Administrador
Nombre historia: Diseñar interfaz	
Estimación inicial: 6	Iteración asignada: 4
Descripción: La interfaz del sistema debe tener un diseño sencillo, con colores pocos llamativos. Debe mostrar información relacionada solamente con las funcionalidades del sistema, para no facilitar distracción en el trabajo que se desarrolle.	
Cómo probarlo:	
Notas:	

Tabla 2.22 Historia técnica Diseñar la ayuda del sistema.

Historia de Usuario	
Número: 21	Usuario: Administrador
Nombre historia: Diseñar la ayuda del sistema.	
Estimación inicial: 6	Iteración asignada: 4
Descripción: El sistema debe proporcionar materiales de ayuda para hacer mejor uso de los servicios que este ofrece. Debe ser diseñada sobre la premisa de que los usuarios finales del mismo no poseen conocimientos avanzados en la informática y en navegación de sitios web.	
Cómo probarlo:	
Notas:	

2.4.6 Requerimientos adicionales

Existen requerimientos adicionales que el sistema debe tener asociados al cumplimiento de las funcionalidades del mismo:

- Rendimiento
 - Estará implementado sobre una tecnología Web, facilitando su uso a través de la red.
 - El sistema deberá ser capaz de gestionar toda la información y dar respuesta en el menor tiempo posible.
- Seguridad.
 - Solo tendrán acceso al sistema aquellos usuarios que tengan permiso determinados por el Especialista de Seguridad y Protección.
 - Chequeo de seguridad sobre las acciones tales como verificación de borrado, de modificado, etc.
- Hardware.
 - Se requiere de una impresora estándar para la impresión del “Reporte Semanal” y el “Informe Mensual”.
- Software.
 - Apache como servidor de aplicaciones.
 - MySQL como gestor de base de datos.
- Restricciones en el diseño.
 - Para organizar el diseño del sistema, debe ser utilizada la metodología Scrum y XP, usando el lenguaje de modelación UML y como herramienta para llevarlo a cabo el Visual Paradigm.
- Legales.

Debe cumplir con lo estipulados en:

- Decreto Ley – 186 / 98.
- Manual de Procedimientos. Tema 01.04.00.06. Elaboración de informes y documentos. Procedimiento 01.04.00.06.01: Sistema informativo regular.
- Manual de Procedimientos. Tema 01.04.00.02. Enfrentamiento al delito. Procedimiento 01.04.00.02.01.: Presencia de un posible o real hecho delictivo. Procedimiento 01.04.00.02.02.: Control de los delitos.

2.4.7 Planificación de los Sprints

Cada *sprint* tiene una reunión inicial de planificación en la que se decide que historias se van a incluir en el mismo.

El *Sprint Backlog* es la lista de tareas en las que se dividen las historias, con el fin de convertirlas en funcionalidades del sistema, logrando que incremente el desarrollo del producto según se vayan elaborando cada uno de los *Sprints*. Son una especie de guía, que el equipo confecciona con el firme propósito de orientar sus trabajos mientras implementan el sistema.

Para una correcta planificación de los *sprints* son necesarios una serie de datos:

- **Cantidad de días-hombre:** Es la suma de los días que puede dedicar cada miembro del equipo de trabajo a la producción durante el *sprint*.
- **Factor de dedicación del equipo:** Es una estimación del aprovechamiento del tiempo por parte del equipo de trabajo durante el *sprint*. Se representa en %.
- **Velocidad estimada del sprint:** Se calcula multiplicando la cantidad de días – hombre por el factor de dedicación.
- **Velocidad final del sprint:** Es la suma de las estimaciones iniciales de las historias de usuarios que se incluyeron en el *sprint*.

A continuación aparecen las planificaciones de cada uno de los *Sprint* creados para construir la aplicación:

Sprint 1 (Crear Contenido)

Cantidad de días que debe durar el *sprint*: 56. Del 13/10/2008 al 8/12/2008

Cantidad de días – hombre: Son dos miembros en el equipo. Cada uno puede aportar en ese período un total de 25 días. Esto hace un total de 50 días – hombre.

Factor de dedicación: 85%.

Velocidad estimada: $50 \times 85\% = 42.5$ puntos – historia. Esto significa que en este *sprint* solo se pueden acumular historias de usuario hasta que la suma de sus estimaciones esté por debajo de 42.5.

Las historias seleccionadas para este *sprint* son las siguientes:

- 1- Historia de usuario Reportar una Incidencia.
- 2- Historia de usuario Crear un Expediente.
- 3- Historia de usuario Adicionar un Infractor.
- 4- Historia de usuario Crear un Reporte Semanal.
- 5- Historia de usuario Crear un Informe Mensual.

En total, las estimaciones de las 5 historias de usuario suman 40 puntos. Por tanto, la velocidad final del *sprint* es de 40 puntos.

Sprint Backlog 1:

Tabla 2.23 Tarea Crear formulario Reportar Incidencia.

Tarea	
Número tarea: 1	Nombre de la historia: Reportar una Incidencia
Nombre tarea: Crear formulario Reportar Incidencia	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 13/10/2008	Fecha fin: 18/10/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un formulario que recoja los datos relevantes de una incidencia en tres secciones: Datos generales, Datos del informante y Datos del detector.	

Tabla 2.24 Tarea Validar campos numéricos y obligatorios del formulario Reportar Incidencia.

Tarea	
Número tarea: 2	Nombre de la historia: Reportar una Incidencia
Nombre tarea: Validar campos numéricos y obligatorios del formulario Reportar Incidencia	
Tipo de tarea: Desarrollo	Estimación inicial: 4
Fecha inicio: 19/10/2008	Fecha fin: 21/10/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Definir como campos obligatorios del formulario: asunto y quién recibe el reporte; en la sección Datos generales: fecha en que se detectó y descripción de la incidencia; en la sección Datos del detector: nombre y primer apellido y en la sección Datos del informante: nombre, primer apellido, solapín, carnet de identidad del informante y la fecha y hora en que se reportó la incidencia. Definir como campos numéricos el carnet de identidad del detector y del informante.	

Tabla 2.25 Tarea Elaborar formulario Crear Expediente.

Tarea	
Número tarea: 3	Nombre de la historia: Crear un Expediente
Nombre tarea: Elaborar formulario Crear Expediente	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 22/10/2008	Fecha fin: 26/10/2008

Programador responsable: Manuel Quert - Migdelys Sablón
Descripción: Crear un formulario que recoja los datos relevantes de un expediente en cinco secciones: Datos generales, Detalles, Valoración material, Otros e Infractores.

Tabla 2.26 Tarea Validar campos obligatorios y numéricos del formulario Crear Expediente.

Tarea	
Número tarea: 4	Nombre de la historia: Crear un Expediente
Nombre tarea: Validar campos obligatorios y numéricos del formulario Crear Expediente	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 27/10/2008	Fecha fin: 1/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Definir como campos obligatorios del formulario: elaborado por y tipificación; en la sección Datos generales: número del expediente, fecha y hora de elaboración y estado; en la sección Detalles: lugar del hecho y descripción; en la sección Valoración material: medios afectados y en la sección Otros: protección del lugar, esclarecimiento y clasificación. Definir como campos numéricos el número del expediente.	

Tabla 2.27 Tarea Crear formulario Adicionar Infractor.

Tarea	
Número tarea: 5	Nombre de la historia: Adicionar un Infractor
Nombre tarea: Crear formulario Adicionar Infractor	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 2/11/2008	Fecha fin: 7/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un formulario que recoja los datos de un infractor.	

Tabla 2.28 Tarea Validar campos numéricos y obligatorios del formulario Adicionar Infractor.

Tarea	
Número tarea: 6	Nombre de la historia: Adicionar un Infractor

Nombre tarea: Validar campos numéricos y obligatorios del formulario Adicionar Infractor	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 8/11/2008	Fecha fin: 13/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Definir como campos obligatorios del formulario: nombre y apellidos y carnet de identidad. Definir como campo numérico: carnet de identidad.	

Tabla 2.29 Tarea Elaborar formulario Crear Reporte Semanal.

Tarea	
Número tarea: 7	Nombre de la historia: Crear un Reporte Semanal
Nombre tarea: Elaborar formulario Crear Reporte Semanal	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 14/11/2008	Fecha fin: 19/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un formulario que recoja los datos de un reporte semanal.	

Tabla 2.30 Tarea Validar campos obligatorios del formulario Crear Reporte Semanal.

Tarea	
Número tarea: 8	Nombre de la historia: Crear un Reporte Semanal
Nombre tarea: Validar campos obligatorios del formulario Crear Reporte Semanal	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 19/11/2008	Fecha fin: 24/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Definir como campos obligatorios del formulario: semana, fecha de inicio y fecha final.	

Tabla 2.31 Tarea Elaborar formulario Crear Informe Mensual.

Tarea	
Número tarea: 9	Nombre de la historia: Crear un Informe Mensual

Nombre tarea: Elaborar formulario Crear Informe Mensual	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 25/11/2008	Fecha fin: 30/11/2008
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un formulario que recoja los datos de un informe mensual.	

Tabla 2.32 Tarea Validar campos obligatorios del formulario Crear Informe Mensual.

		Tarea
Número tarea: 10	Nombre de la historia: Crear un Informe Mensual	
Nombre tarea: Validar campos obligatorios del formulario Crear Informe Mensual		
Tipo de tarea : Desarrollo	Estimación inicial: 4	
Fecha inicio: 1/12/2008	Fecha fin: 5/12/2008	
Programador responsable: Manuel Quert - Migdelys Sablón		
Descripción: Definir como campos obligatorios del formulario: encabezado del informe, mes a informar, fecha inicial y fecha final.		

En la figura 2 se muestra el diagrama Burndown chart del *Sprint* 1, el mismo representa el comportamiento del *sprint* respecto al cumplimiento de las tareas:

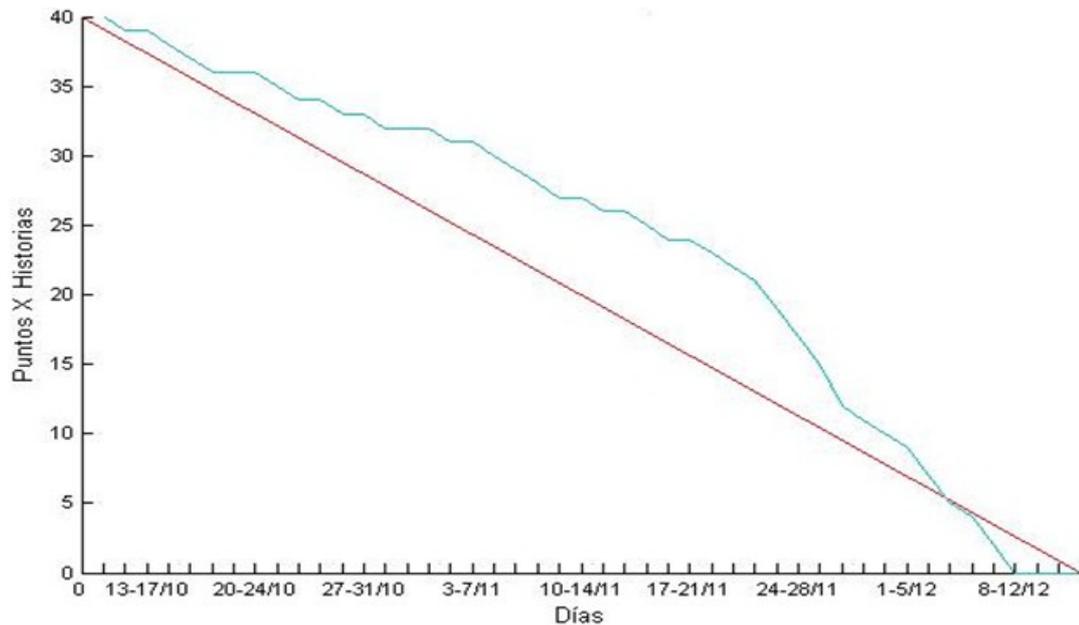


Figura 2. Diagrama Burndown chart *Sprint 1*.

Al término del *Sprint 1* (Crear Contenido) se completa el 15% de las funcionalidades del sistema. Con el sistema en este estado, los usuarios finales pueden crear, expedientes, incidencias, infractores, reportes semanales e informes mensuales.

Sprint 2 (Buscar, modificar y eliminar contenido)

Cantidad de días que debe durar el *sprint*: 49. Del 7/1/2009 al 13/2/2009

Cantidad de días – hombre: Son dos miembros en el equipo. Cada uno puede aportar en ese período un total de 35 días. Esto hace un total de 70 días – hombre.

Factor de dedicación: 90%.

Velocidad estimada: $70 \times 90\% = 63$ puntos – historia. Esto significa que en este *sprint* solo se pueden acumular historias de usuario hasta que la suma de sus estimaciones esté por debajo de 63.

Las historias seleccionadas para este *sprint* son las siguientes:

- 1- Historia de usuario Buscar Contenido.
- 2- Historia de usuario Modificar una Incidencia.
- 3- Historia de usuario Modificar un Expediente.
- 4- Historia de usuario Modificar un Infractor.

- 5- Historia de usuario Modificar un Reporte Semanal.
- 6- Historia de usuario Modificar un Informe Mensual.
- 7- Historia de usuario Eliminar una Incidencia.
- 8- Historia de usuario Eliminar un Expediente.
- 9- Historia de usuario Eliminar un Infractor.
- 10- Historia de usuario Eliminar un Reporte Semanal.
- 11- Historia de usuario Eliminar un Informe Mensual.

En total, las estimaciones de las 11 historias de usuario suman 60 puntos. Por tanto, la velocidad final del *sprint* es de 60 puntos.

Sprint Backlog 2:

Tabla 2.33 Tarea Crear buscador de Incidencias.

Tarea	
Número tarea: 11	Nombre de la historia: Buscar Contenido
Nombre tarea: Crear buscador de Incidencias	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 7/1/2009	Fecha fin: 9/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un buscador para el contenido Incidencia que permita filtrar por: quién reportó la incidencia, la descripción, día, mes y año en que fue reportada.	

Tabla 2.34 Tarea Crear buscador de Expedientes.

Tarea	
Número tarea: 12	Nombre de la historia: Buscar Contenido
Nombre tarea: Crear buscador de Expedientes	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 10/1/2009	Fecha fin: 12/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un buscador para el contenido Expediente que permita filtrar por: estado, quién lo elaboró, número del expediente, lugar del hecho, año, mes, día y clasificación.	

Tabla 2.35 Tarea Crear buscador de Infractores.

Tarea	
Número tarea: 13	Nombre de la historia: Buscar Contenido
Nombre tarea: Crear buscador de Infractores	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 13/1/2009	Fecha fin: 15/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un buscador para el contenido Infractor que permita filtrar por: nombres y apellidos, carnet de identidad y ocupación.	

Tabla 2.36 Tarea Crear buscador de Reportes Semanales.

Tarea	
Número tarea: 14	Nombre de la historia: Buscar Contenido
Nombre tarea: Crear buscador de Reportes Semanales	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 16/1/2009	Fecha fin: 18/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un buscador para el contenido Reporte semanal que permita filtrar por el nombre de la semana.	

Tabla 2.37 Tarea Crear buscador de Informes Mensuales.

Tarea	
Número tarea: 15	Nombre de la historia: Buscar Contenido
Nombre tarea: Crear buscador de Informes Mensuales	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 19/1/2009	Fecha fin: 21/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear un buscador para el contenido Informe Mensual que permita filtrar por el mes a informar.	

Tabla 2.38 Tarea Crear enlace Editar a los formularios de las Incidencias.

Tarea	
Número tarea: 16	Nombre de la historia: Modificar una Incidencia
Nombre tarea: Crear enlace Editar a los formularios de las Incidencias	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 22/1/2009	Fecha fin: 24/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de incidencias un hipervínculo al formulario de la incidencia seleccionada para modificar sus datos. Los datos modificados se guardan mediante el <i>submit</i> del formulario.	

Tabla 2.39 Tarea Crear enlace Editar a los formularios de los Expedientes.

Tarea	
Número tarea: 17	Nombre de la historia: Modificar un Expediente
Nombre tarea: Crear enlace Editar a los formularios de los Expedientes	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 25/1/2009	Fecha fin: 27/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de expedientes un hipervínculo al formulario del expediente seleccionado para modificar sus datos. Los datos modificados se guardan mediante el <i>submit</i> del formulario.	

Tabla 2.40 Tarea Crear enlace Modificar a los formularios de los Infractores.

Tarea	
Número tarea: 18	Nombre de la historia: Modificar un Infractor
Nombre tarea: Crear enlace Modificar a los formularios de los Infractores	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 27/1/2009	Fecha fin: 28/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de infractores un hipervínculo al formulario del infractor seleccionado para modificar sus datos. Los datos modificados se guardan mediante el <i>submit</i> del formulario.	

Tabla 2.41 Tarea Crear enlace Editar a los formularios de los Reportes Semanales.

Tarea	
Número tarea: 19	Nombre de la historia: Modificar Reporte Semanal
Nombre tarea: Crear enlace Editar a los formularios de los Reportes Semanales	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 29/1/2009	Fecha fin: 30/1/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de reportes semanales un hipervínculo al formulario del reporte semanal seleccionado para modificar sus datos. Los datos modificados se guardan mediante el <i>submit</i> del formulario.	

Tabla 2.42 Tarea Crear enlace Editar a los formularios de los Informes Mensuales.

Tarea	
Número tarea: 20	Nombre de la historia: Modificar Informe Mensual
Nombre tarea: Crear enlace Editar a los formularios de los Informes Mensuales	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 31/1/2009	Fecha fin: 1/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de informes mensuales un hipervínculo al formulario del informe mensual seleccionado para modificar sus datos. Los datos modificados se guardan mediante el <i>submit</i> del formulario.	

Tabla 2.43 Tarea Crear enlace Borrar a los formularios de las Incidencias.

Tarea	
Número tarea: 21	Nombre de la historia: Eliminar una Incidencia
Nombre tarea: Crear enlace Borrar a los formularios de las Incidencias	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 2/2/2009	Fecha fin: 3/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de incidencias un hipervínculo que permita borrar la incidencia seleccionada. Se debe solicitar la confirmación del usuario antes de eliminar la incidencia.	

CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

Tabla 2.44 Tarea Crear enlace Borrar a los formularios de los Expedientes.

Tarea	
Número tarea: 22	Nombre de la historia: Eliminar un Expediente
Nombre tarea: Crear enlace Borrar a los formularios de los Expedientes	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 4/2/2009	Fecha fin: 5/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de expedientes un hipervínculo que permita borrar el expediente seleccionado. Se debe solicitar la confirmación del usuario antes de eliminar el expediente.	

Tabla 2.45 Tarea Crear enlace Eliminar a los formularios de los Infractores.

Tarea	
Número tarea: 23	Nombre de la historia: Eliminar un Infractor
Nombre tarea: Crear enlace Eliminar a los formularios de los Infractores	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 6/2/2009	Fecha fin: 7/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de infractores un hipervínculo que permita borrar el infractor seleccionado. Se debe solicitar la confirmación del usuario antes de eliminar el infractor.	

Tabla 2.46 Tarea Crear enlace Borrar a los formularios de los Reportes Semanales.

Tarea	
Número tarea: 24	Nombre de la historia: Eliminar un Reporte Semanal
Nombre tarea: Crear enlace Borrar a los formularios de los Reportes Semanales	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 8/2/2009	Fecha fin: 9/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de reportes semanales un hipervínculo que permita borrar el reporte seleccionado. Se debe solicitar la confirmación del usuario antes de eliminar el reporte.	

Tabla 2.47 Tarea Crear enlace Borrar a los formularios de los Informes Mensuales.

Tarea	
Número tarea: 25	Nombre de la historia: Eliminar un Informe Mensual
Nombre tarea: Crear enlace Borrar a los formularios de los Informes Mensuales	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 10/2/2009	Fecha fin: 11/2/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Insertar en el buscador de informes mensuales un hipervínculo que permita borrar el informe seleccionado. Se debe solicitar la confirmación del usuario antes de eliminar el informe.	

En la figura 3 se muestra el diagrama Burndown chart del *Sprint 2*, el mismo representa el comportamiento del *sprint* respecto al cumplimiento de las tareas:

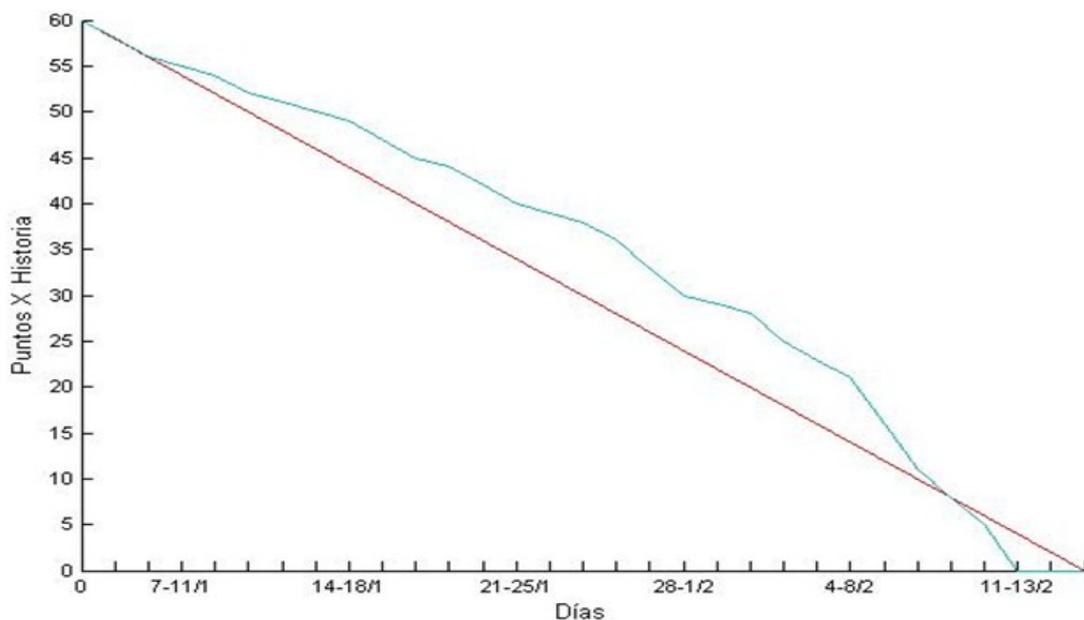


Figura 3. Diagrama Burndown chart *Sprint 2*.

Con la incorporación al sistema de las funcionalidades para la búsqueda y manejo de contenido en el *Sprint II*, el producto está en un 55% de terminación. Los reportes permiten mantener actualizaciones de la situación general en cada semana y en cada mes. El control de infractores, por su parte, es vital a la hora

de reubicar trabajadores con antecedentes de delitos en la Universidad o de tomar medidas disciplinarias con los mismos. Estas funcionalidades garantizan un control estadístico de las incidencias y los expedientes, aportando un importante valor añadido al sistema.

Sprint 3 (Implementación de las vistas PDF)

Cantidad de días que debe durar el *sprint*: 32. Del 23/2/2009 al 27/3/2009

Cantidad de días – hombre: Son dos miembros en el equipo. Cada uno puede aportar en ese período un total de 25 días. Esto hace un total de 50 días – hombre.

Factor de dedicación: 85%.

Velocidad estimada: $50 \times 85\% = 42.5$ puntos – historia. Esto significa que en este *sprint* solo se pueden acumular historias de usuario hasta que la suma de sus estimaciones esté por debajo de 42.5. La historia seleccionada para este *sprint* es:

- 1- Historia de usuario Mostrar Contenido.

En total, la estimación de la historia de usuario es de 40 puntos. Por tanto, la velocidad final del *sprint* es de 40 puntos.

Sprint Backlog 3:

Tabla 2.48 Tarea Implementar la vista de las incidencias en formato PDF.

Tarea	
Número tarea: 26	Nombre de la historia: Mostrar Contenido
Nombre tarea: Implementar la vista de las incidencias en formato PDF	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 23/2/2009	Fecha fin: 1/3/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Implementar un <i>script</i> de extensión PHP para la creación dinámica de una vista PDF de las incidencias. Insertar en el buscador de incidencias un hipervínculo hacia el <i>script</i> antes mencionado y mostrar los datos de la incidencia seleccionada en formato PDF.	

Tabla 2.49 Tarea Implementar la vista de los expedientes en formato PDF.

Tarea	
Número tarea: 27	Nombre de la historia: Mostrar Contenido
Nombre tarea: Implementar la vista de los expedientes en formato PDF	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 2/3/2009	Fecha fin: 8/3/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Implementar un <i>script</i> de extensión PHP para la creación dinámica de una vista PDF de los expedientes. Insertar en el buscador de expedientes un hipervínculo hacia el <i>script</i> antes mencionado y mostrar los datos del expediente seleccionado en formato PDF.	

Tabla 2.50 Tarea Implementar la vista de los Infractores en formato PDF.

Tarea	
Número tarea: 28	Nombre de la historia: Mostrar Contenido
Nombre tarea: Implementar la vista de los Infractores en formato PDF	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 9/3/2009	Fecha fin: 14/3/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Implementar un <i>script</i> de extensión PHP para la creación dinámica de una vista PDF del historial de los infractores. Insertar en el buscador de infractores un hipervínculo hacia el <i>script</i> antes mencionado y mostrar los datos del historial seleccionado en formato PDF.	

Tabla 2.51 Tarea Implementar la vista de los Reportes Semanales en formato PDF.

Tarea	
Número tarea: 29	Nombre de la historia: Mostrar Contenido
Nombre tarea: Implementar la vista de los Reportes Semanales en formato PDF	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 15/3/2009	Fecha fin: 20/3/2009
Programador responsable: Manuel Quert - Migdelys Sablón	

<p>Descripción: Implementar un <i>script</i> de extensión PHP para la creación dinámica de una vista PDF de los reportes semanales. Insertar en el buscador de reportes semanales un hipervínculo hacia el <i>script</i> antes mencionado y mostrar los datos del reporte semanal seleccionado en formato PDF.</p>
--

Tabla 2.52 Tarea Implementar la vista de los Informes Mensuales en formato PDF.

Tarea	
Número tarea: 30	Nombre de la historia: Mostrar Contenido
Nombre tarea: Implementar la vista de los Informes Mensuales en formato PDF	
Tipo de tarea : Desarrollo	Estimación inicial: 8
Fecha inicio: 21/3/2009	Fecha fin: 26/3/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
<p>Descripción: Implementar un <i>script</i> de extensión PHP para la creación dinámica de una vista PDF de los informes mensuales. Insertar en el buscador de reportes semanales un hipervínculo hacia el <i>script</i> antes mencionado y mostrar los datos del informe mensual seleccionado en formato PDF.</p>	

En la figura 4 se muestra el diagrama Burndown chart del *Sprint* 3, el mismo representa el comportamiento del *sprint* respecto al cumplimiento de las tareas:

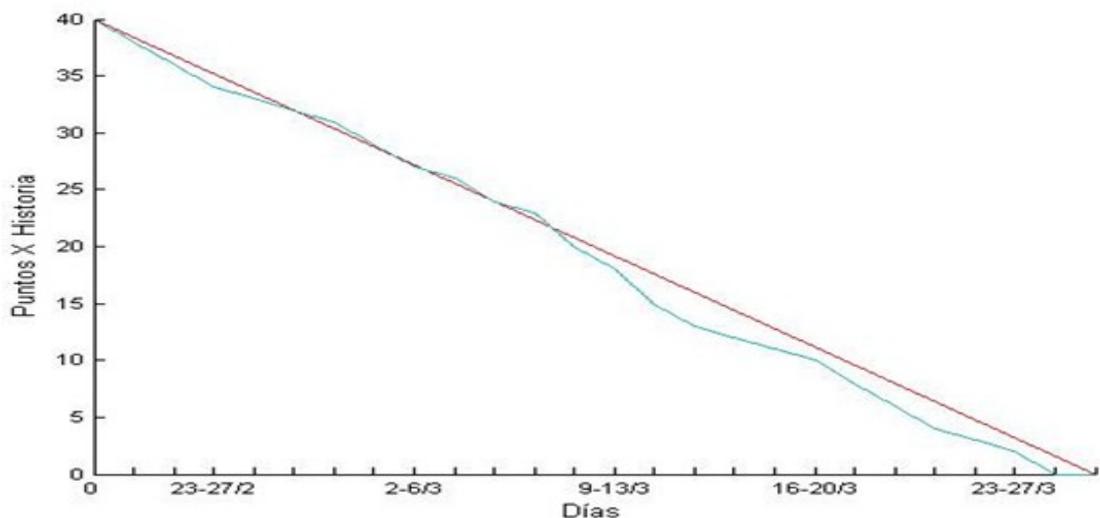


Figura 4. Diagrama Burndown chart *Sprint* 3.

En el *Sprint* III se implementa la generación dinámica de documentos PDF. Estos documentos complementan las funcionalidades de reportes al incluir las opciones de copia e impresión. Al término de este *Sprint*, los procesos de gestión funcionan íntegramente y el sistema se encuentra en un 80% de terminación.

Sprint 4 (Configuración de la interfaz y seguridad del sistema)

Cantidad de días que debe durar el *sprint*: 12. Del 9/4/2009 al 20/4/2009

Cantidad de días – hombre: Son dos miembros en el equipo. Cada uno puede aportar en ese período un total de 12 días. Esto hace un total de 24 días – hombre.

Factor de dedicación: 85%.

Velocidad estimada: $24 \times 85\% = 20.4$ puntos – historia. Esto significa que en este *sprint* solo se pueden acumular historias de usuario hasta que la suma de sus estimaciones esté por debajo de 20.4.

Las historias seleccionadas para este *sprint* son las siguientes:

- 1- Historia de usuario Autenticar Usuarios.
- 2- Historia de usuario Administrar Usuarios.
- 3- Historia de usuario Diseñar interfaz.
- 4- Historia de usuario Diseñar la ayuda del sistema.

En total, las estimaciones de las historias de usuario son de 2.0 puntos. Por tanto, la velocidad final del *sprint* es de 20 puntos.

Sprint Backlog 4:

Tabla 2.53 Tarea Configurar la autenticación en el sistema.

		Tarea
Número tarea: 31	Nombre de la historia: Autenticar Usuarios	
Nombre tarea: Configurar la autenticación en el sistema por el dominio de la Universidad		
Tipo de tarea : Desarrollo	Estimación inicial: 4	
Fecha inicio: 9/4/2009	Fecha fin: 11/4/2009	
Programador responsable: Manuel Quert - Migdelys Sablón		
Descripción: Configuración del módulo LDAP con los parámetros de conexión de la universidad.		

Tabla 2.54 Tarea Crear y configurar el rol Técnico.

Tarea	
Número tarea: 32	Nombre de la historia: Administrar Usuarios
Nombre tarea: Crear y configurar el rol Técnico	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 12/4/2009	Fecha fin: 13/4/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear el rol Técnico con permisos limitados al reporte y búsqueda de incidencias. Es importante que este rol solo pueda modificar información de una incidencia antes de enviarla por primera vez.	

Tabla 2.55 Tarea Crear y configurar el rol Especialista.

Tarea	
Número tarea: 33	Nombre de la historia: Administrar Usuarios
Nombre tarea: Crear y configurar el rol Especialista	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 14/4/2009	Fecha fin: 15/4/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
Descripción: Crear el rol Especialista con permisos para crear y modificar cualquier contenido almacenado en el sistema. Además podrá asignar los roles de técnico y especialista a los usuarios que estime conveniente.	

Tabla 2.56 Tarea Crear menú principal.

Tarea	
Número tarea: 1	Nombre de la historia: Diseñar interfaz
Nombre tarea: Crear el menú principal	
Tipo de tarea : Desarrollo	Estimación inicial: 4
Fecha inicio: 16/4/2009	Fecha fin: 17/4/2009
Programador responsable: Manuel Quert - Migdelys Sablón	

<p>Descripción: Creación del menú principal que cuenta con cinco secciones: Incidencias, Expedientes, Infractores, Reportes e Informes y Administrar. Las primeras 4 secciones permiten crear y buscar el contenido al cual hacen referencia y la última sección es la encargada de configurar y administrar el sistema.</p>
--

Tabla 2.57 Tarea Seleccionar y configurar la interfaz.

Tarea	
Número tarea: 2	Nombre de la historia: Diseñar interfaz
Nombre tarea: Seleccionar y configurar la interfaz	
Tipo de tarea : Desarrollo	Estimación inicial: 2
Fecha inicio: 18/4/2009	Fecha fin: 19/4/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
<p>Descripción: Selección de un tema de Drupal. Configurar los aspectos de: nombre del sitio, idioma, página principal y colores.</p>	

Tabla 2.58 Tarea Configuración e implementación de la ayuda.

Tarea	
Número tarea: 3	Nombre de la historia: Diseñar la ayuda del sistema
Nombre tarea: Configuración e implementación de la ayuda	
Tipo de tarea : Desarrollo	Estimación inicial: 6
Fecha inicio: 19/4/2009	Fecha fin: 20/4/2009
Programador responsable: Manuel Quert - Migdelys Sablón	
<p>Descripción: Diseñar el sistema de ayuda al usuario. Desarrollar la versión online y la versión portable.</p>	

En la figura 5 se muestra el diagrama Burndown chart del *Sprint 4*, el mismo representa el comportamiento del *sprint* respecto al cumplimiento de las tareas:

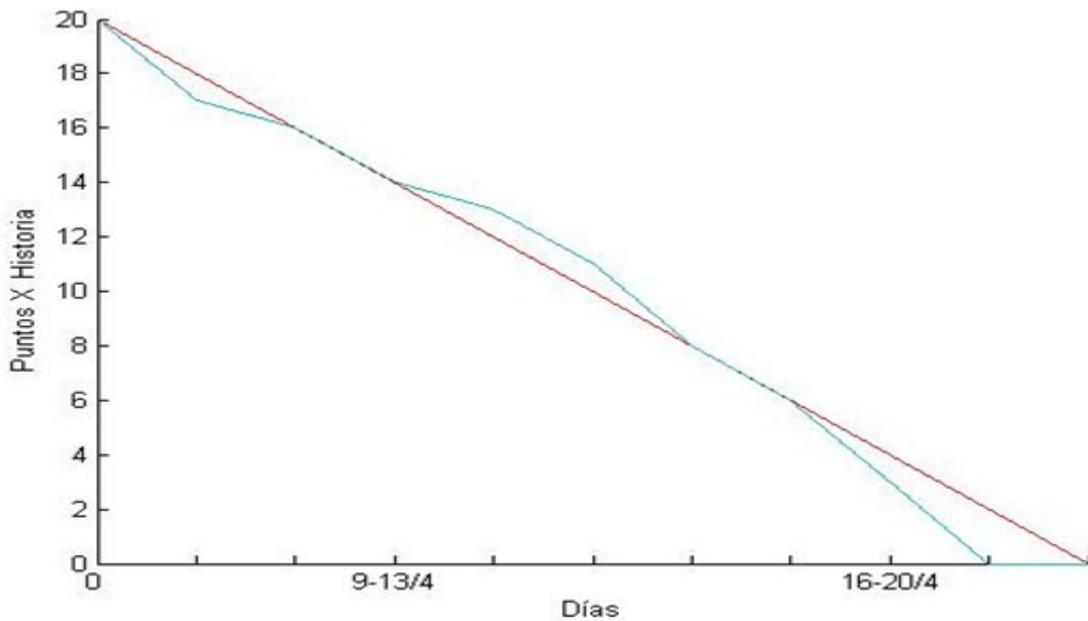


Figura 5. Diagrama Burndown chart *Sprint 4*.

El *Sprint IV* se dedica enteramente a cumplir los requisitos de seguridad e interfaz. Al término del mismo la autenticación de usuarios es por el dominio de la universidad, los roles están bien definidos y operativos y solo pueden autenticarse los usuarios definidos por el Especialista en Seguridad y Protección. En este punto, el sistema está al 100% de terminación.

2.5 Construcción de la propuesta de solución

2.5.1 Diagramas de pantallas por funcionalidades

Los diagramas de pantallas permiten esquematizar al detalle la interfaz gráfica y los contenidos de la aplicación. A continuación se representan los diagramas de pantallas de cada una de las funcionalidades del sistema:

- Reportar Incidencia: Desde la página de inicio se podrá acceder al formulario que contiene los campos que deberán llenarse para crear una incidencia, a través del menú Reportar Incidencia. Después de llenar los campos y enviar la información se mostrará una vista de la incidencia recién guardada. La figura 6 muestra el diagrama de pantallas de la funcionalidad Reportar Incidencia.



Figura 6. Diagrama de pantalla Reportar Incidencia.

- **Crear Reporte Semanal:** Desde la página de inicio se podrá acceder al formulario que contiene los campos que deberán llenarse para crear un reporte semanal, a través del menú Crear Reporte Semanal. Después de llenar los campos y enviar la información se mostrará una vista del reporte recién guardado. La figura 7 muestra el diagrama de pantallas de la funcionalidad Crear Reporte Semanal.



Figura 7. Diagrama de pantalla Crear Reporte Semanal.

- **Crear Informe Mensual:** Desde la página de inicio se podrá acceder al formulario que contiene los campos que deberán llenarse para crear un informe mensual, a través del menú Crear Informe Mensual. Después de llenar los campos y enviar la información se mostrará una vista del informe recién guardado. La figura 8 muestra el diagrama de pantallas de la funcionalidad Crear Informe Mensual.



Figura 8. Diagrama de pantalla Crear Informe Mensual.

- Crear Expediente: Desde la página de inicio se podrá acceder a un formulario que contiene filtros de búsqueda que permitirán buscar la incidencia a la que se le desea crear un expediente, a través del menú Crear Expediente. Una vez encontrada, se accederá al enlace Expediente de la tabla en la que se muestran los datos de las incidencias. El mismo conducirá al formulario que contiene los campos que deberán llenarse para crear el expediente. Después de llenar los campos y enviar la información se mostrará una vista HTML del expediente recién guardado. La figura 9 muestra el diagrama de pantallas de la funcionalidad Crear Expediente.

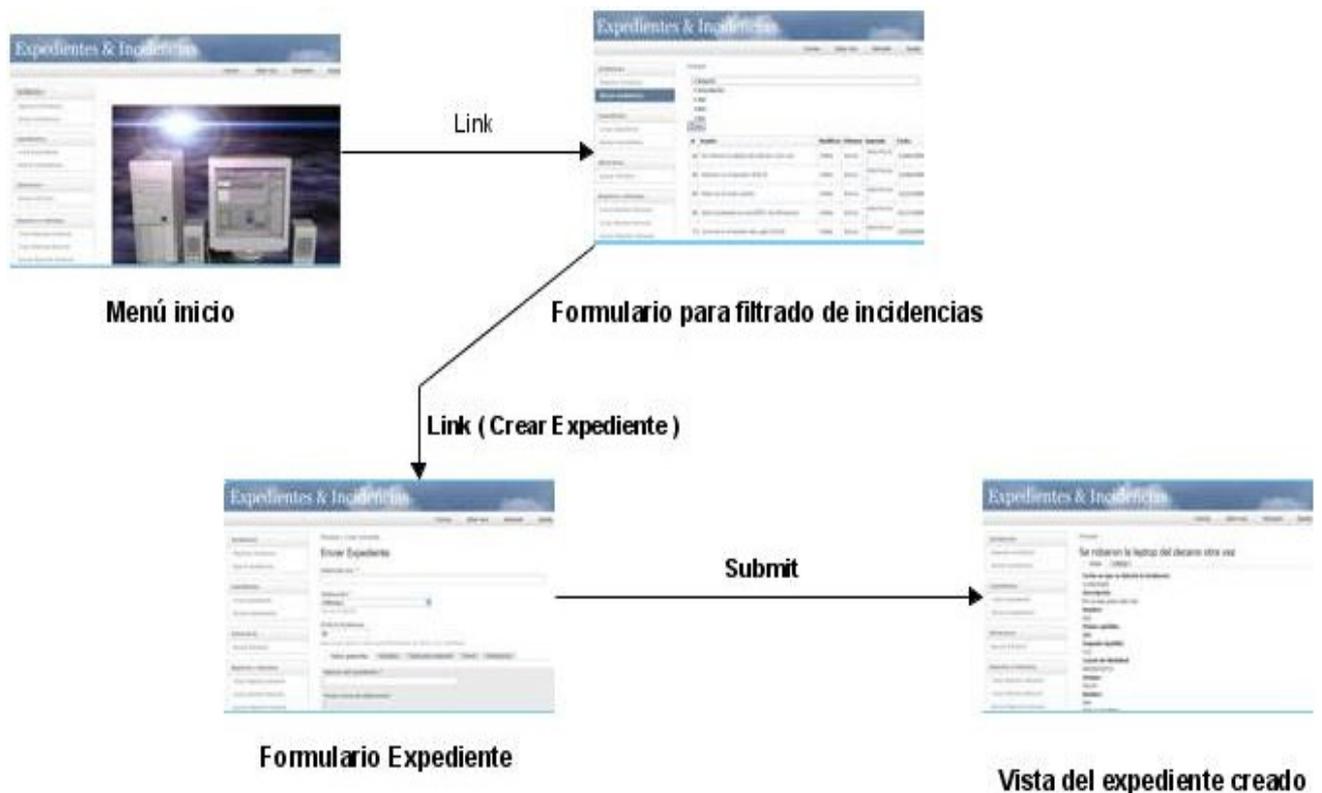


Figura 9. Diagrama de pantalla Crear Expediente.

- **Buscar Incidencias:** Después de acceder al menú Buscar Incidencias desde la página de inicio, el sistema realizará una búsqueda de las incidencias reportadas hasta el momento. Como resultado de la búsqueda se mostrará un formulario que contiene una tabla en la que aparecen enlaces a la información de las incidencias, permitiendo modificarlas, eliminarlas y ver el contenido de las mismas en formato HTML y PDF. La figura 10 muestra el diagrama de pantallas de la funcionalidad Buscar Incidencias.

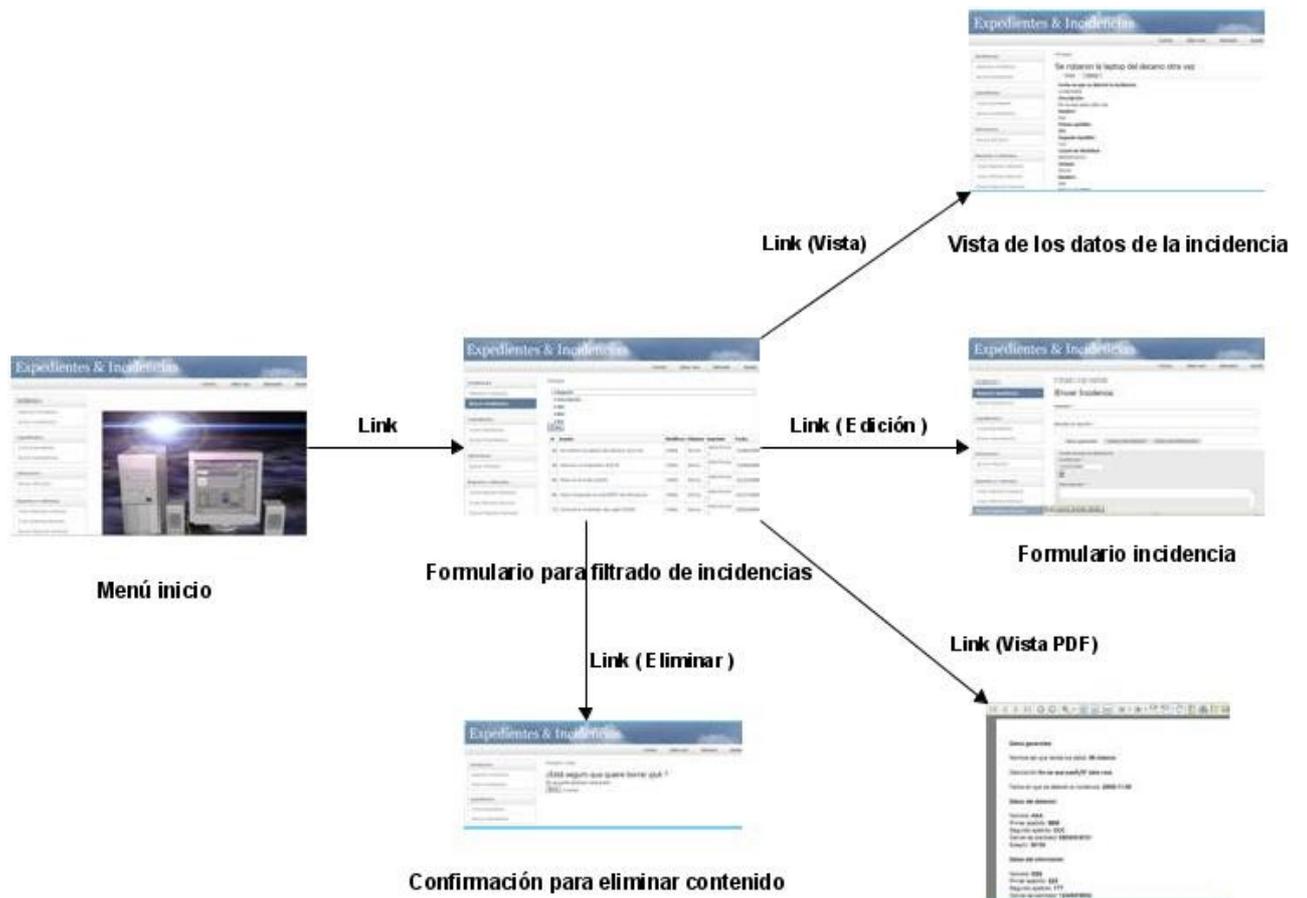


Figura 10. Diagrama de pantalla Buscar Incidencias.

- **Buscar Expedientes:** Después de acceder al menú Buscar Expedientes desde la página de inicio, el sistema realizará una búsqueda de los expedientes elaborados hasta el momento. Como resultado de la búsqueda se mostrará un formulario que contiene una tabla en la que aparecen enlaces a la información de los expedientes, permitiendo modificarlos, eliminarlos y ver el contenido de los mismos en formato HTML y PDF. La figura 11 muestra el diagrama de pantallas de la funcionalidad Buscar Expedientes.

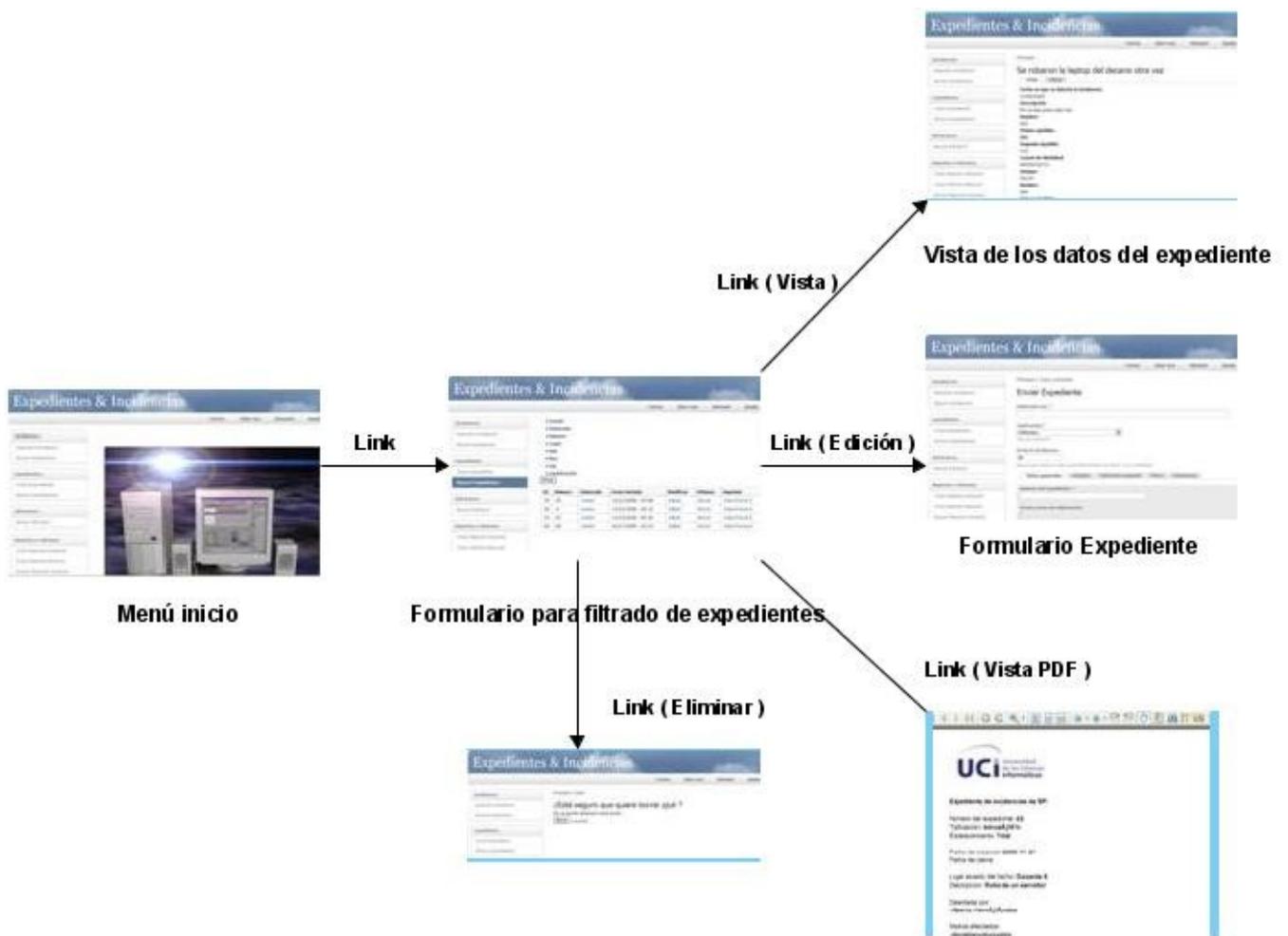


Figura 11. Diagrama de pantalla Buscar Expedientes.

- Adicionar Infractor: Después de buscar y acceder al enlace Editar del expediente al que se desea adicionar un infractor, se mostrará el formulario del expediente con los datos del mismo. En la sección Infractores del expediente se podrá adicionar un infractor conocido o desconocido. En caso de que el infractor ya esté registrado en el sistema, se añade solo el nombre y al enviar los datos se retorna al formulario de búsqueda. En caso contrario, se despliega el formulario que contiene los campos que deberán llenarse para adicionar el infractor. Después de llenar los campos y enviar la información se mostrará una vista HTML de los datos recién enviados. La figura 12 muestra el diagrama de pantallas de la funcionalidad Adicionar Infractor.

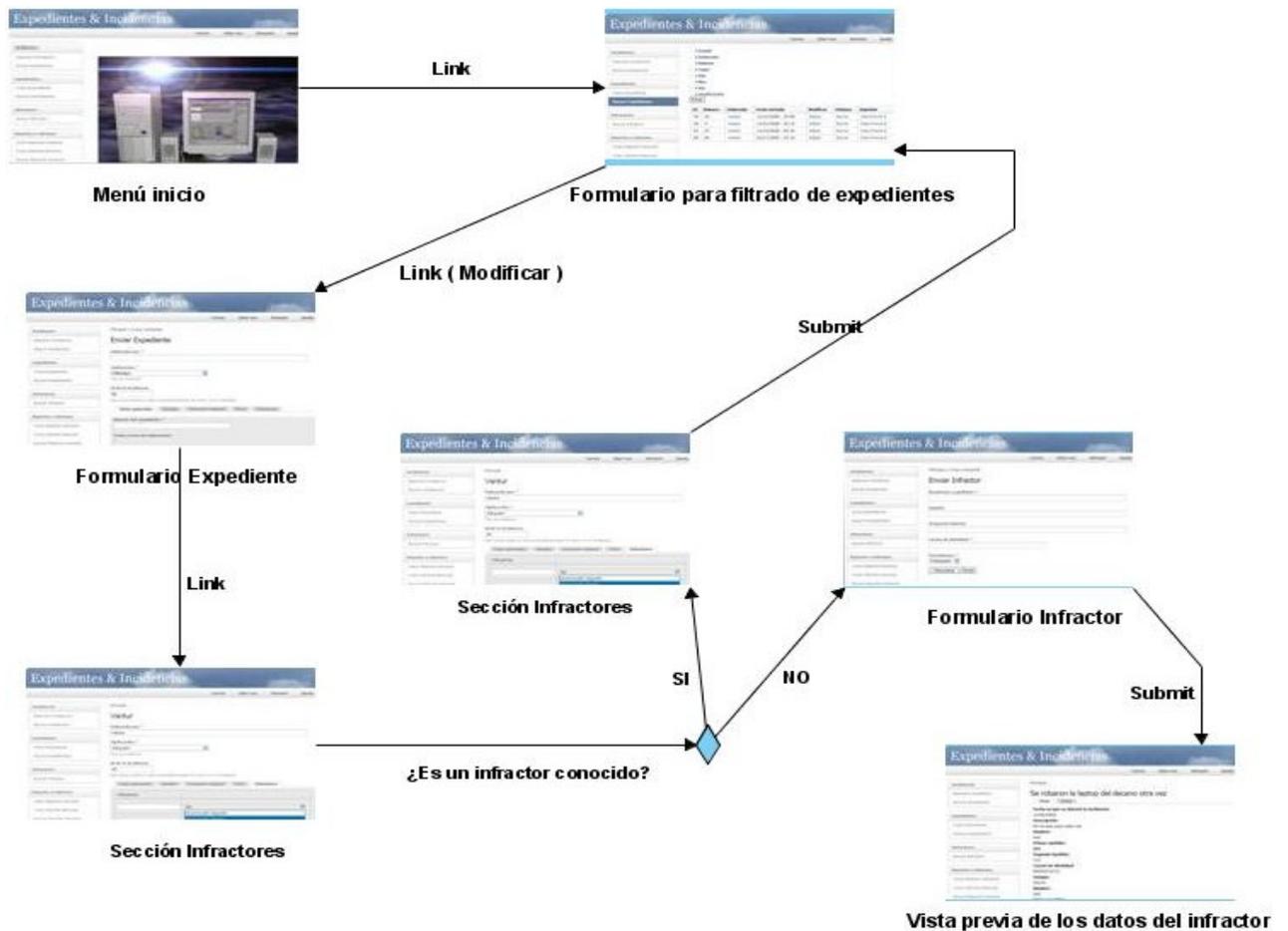


Figura 12. Diagrama de pantalla Adicionar Infractor.

- **Buscar Infractor:** Después de acceder al menú Buscar Infractor desde la página de inicio, el sistema realizará una búsqueda de los infractores insertados en el sistema hasta el momento. Como resultado de la búsqueda se mostrará un formulario que contiene una tabla en la que aparecen enlaces al historial de los infractores, permitiendo modificarlos, eliminarlos y ver el contenido de los mismos en formato HTML y PDF. La figura 13 muestra el diagrama de pantallas de la funcionalidad Buscar Infractor.

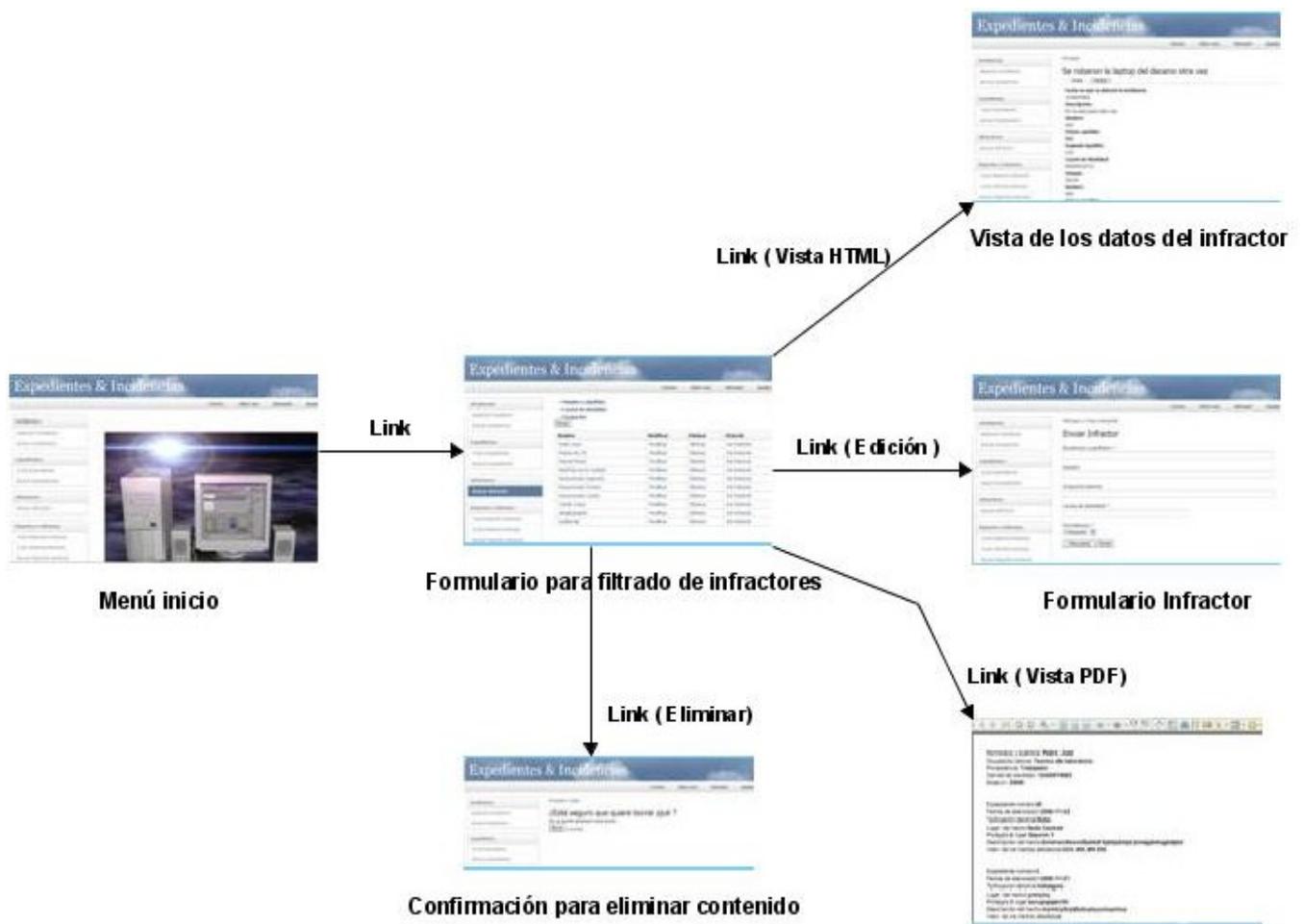


Figura 13. Diagrama de pantalla Buscar Infractor.

- **Buscar Reporte Semanal:** Después de acceder al menú Buscar Reporte Semanal desde la página de inicio, el sistema realizará una búsqueda de los reportes creados hasta el momento. Como resultado de la búsqueda se mostrará un formulario que contiene una tabla en la que aparecen enlaces a la información de los reportes, permitiendo modificarlos, eliminarlos y ver el contenido de los mismos en formato HTML y PDF. La figura 14 muestra el diagrama de pantallas de la funcionalidad Buscar Reporte Semanal.

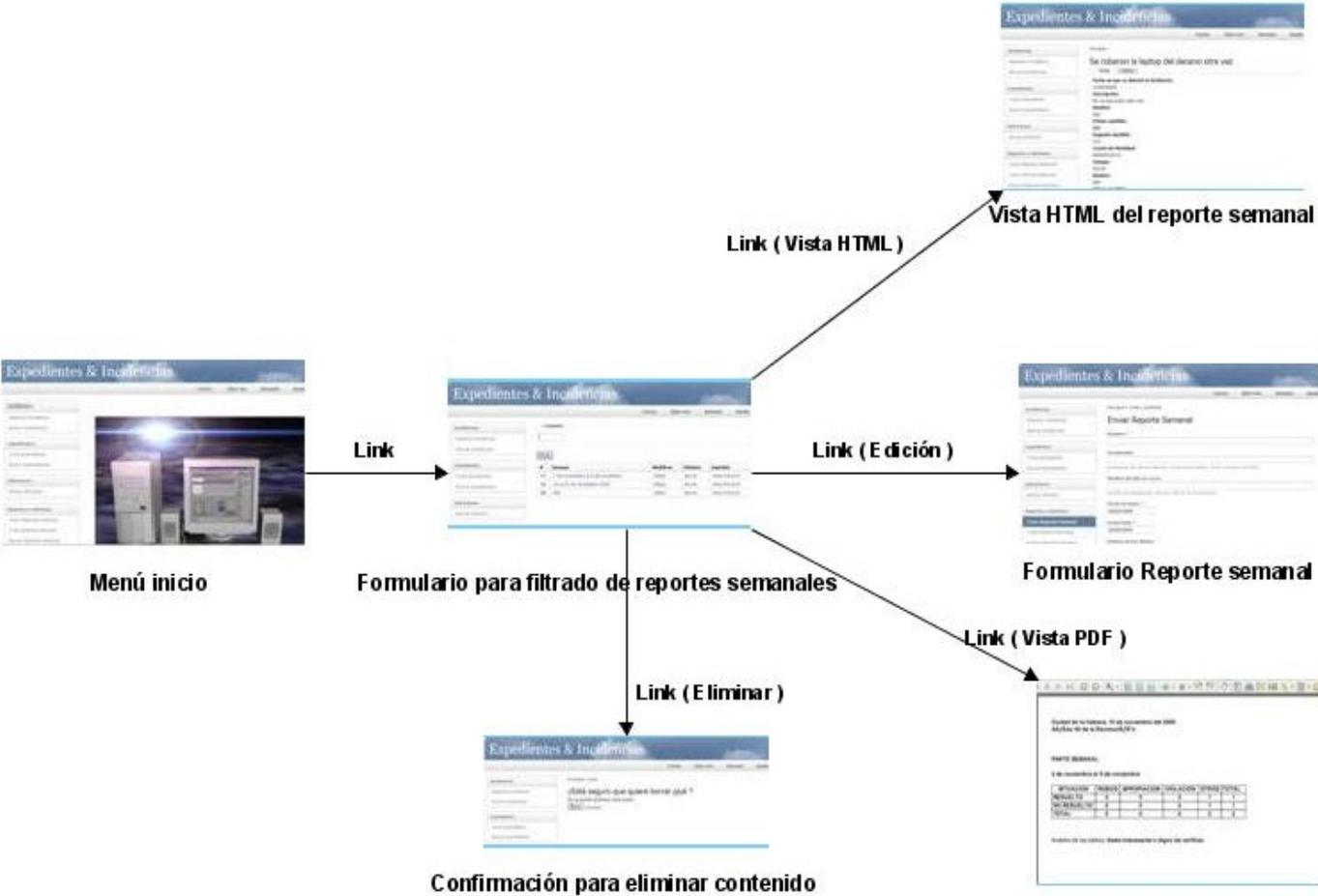


Figura 14. Diagrama de pantalla Buscar Reporte Semanal.

- **Buscar Informe mensual:** Después de acceder al menú Buscar Informe Mensual desde la página de inicio, el sistema realizará una búsqueda de los informes creados hasta el momento. Como resultado de la búsqueda se mostrará un formulario que contiene una tabla en la que aparecen enlaces a la información de los informes, permitiendo modificarlos, eliminarlos y ver el contenido de los mismos en formato HTML y PDF. La figura 15 muestra el diagrama de pantallas de la funcionalidad Buscar Informe mensual.

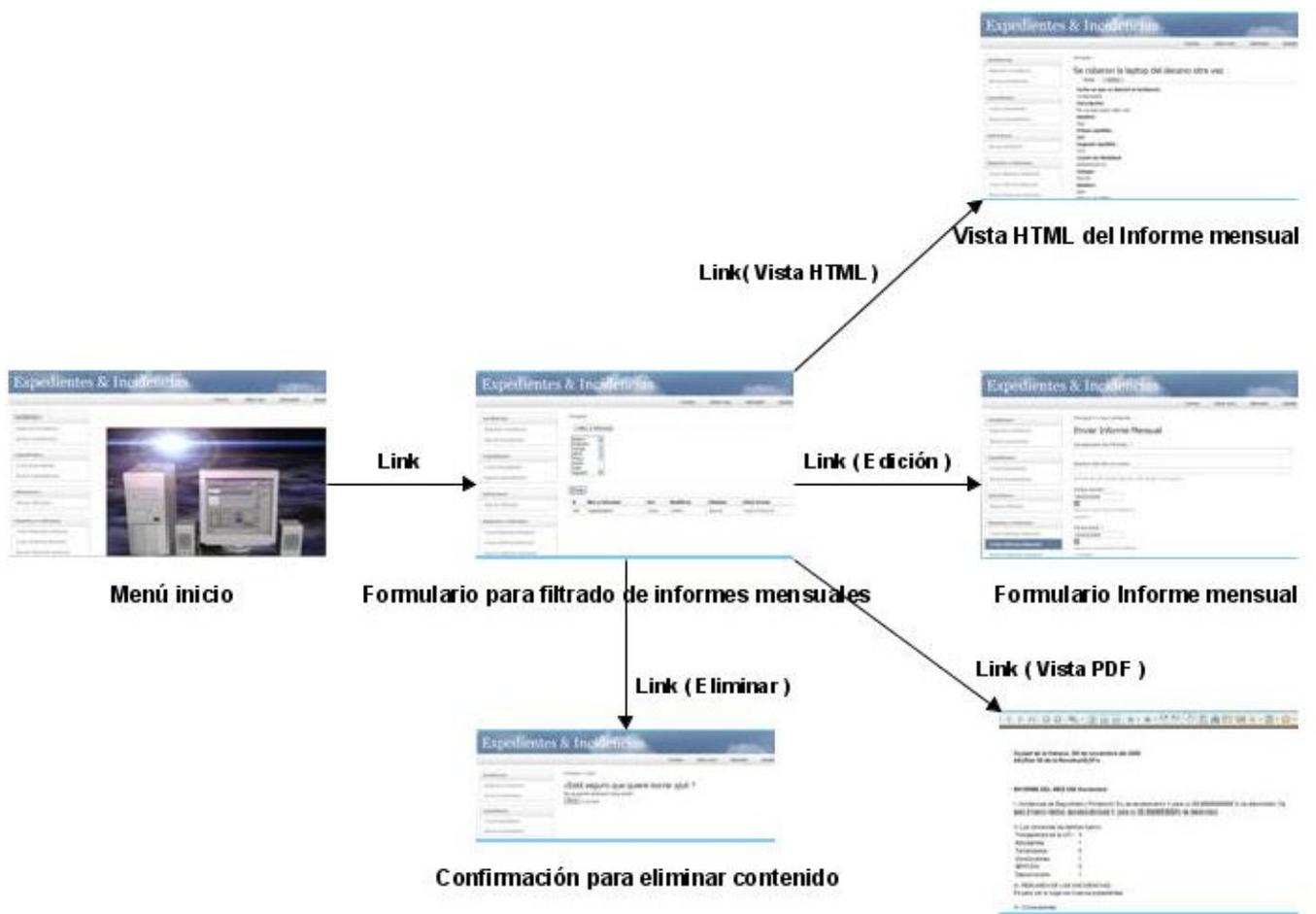


Figura 15. Diagrama de pantalla Buscar Informe Mensual.

2.5.2 Tratamiento de Excepciones

Las excepciones manejadas en los diferentes procesos del sistema, tienen que ver principalmente con la validación de los campos de los formularios. Entre los más comunes figura la omisión de campos obligatorios, introducir una cadena de texto en un campo numérico, escribir direcciones en formatos incorrectos, etc. Esta validación se logra de manera robusta mediante los módulos CCK (*Content Construction Kit*) y JSTools (*Javascript Tools*), los cuales se encargan de la generación de formularios y componentes Javascript.

Los módulos CCK como *Date*, *Number*, *Address*, *Field* y *Text* definen el tipo de los valores que serán entrados por los formularios. Como una validación adicional, el módulo *JavaScript Validator* permite mostrar alertas de error para estos tipos de datos antes que los datos sean enviados al servidor.

2.5.3 Estándares de codificación

Los estándares de codificación son las formas o estilos de programación que utilizan los programadores en la implementación de los diferentes sistemas. Los mismos deben estar establecidos antes de comenzar a desarrollar un proyecto de software, de esta manera se asegura que todos los programadores trabajen de la misma manera y de que si en un futuro se quieren realizar cambios en el código de la aplicación los estándares indicarán como trabajar con el mismo. A continuación se describen algunos de los estándares que utiliza Drupal en el desarrollo de aplicaciones.

- Indentación: Utilizar una indentación de dos espacios sin tabulaciones.
- Estructuras de control de flujo: Incluye las sentencias *if*, *while*, *for*, *switch*. El nombre de la sentencia debe ir separado por un espacio del paréntesis de apertura para evitar confundirlo con las llamadas a funciones. Las llaves (*{ }*) serán siempre utilizadas para delimitar los bloques de instrucciones de las estructuras, incluso cuando su uso sea opcional. De esta manera aumenta la legibilidad del código y disminuyen los errores lógicos al agregarse nuevas instrucciones.

```
if (condicion1 || condicion2) {  
    accion1; }
```

- Llamadas a funciones: No deben existir espacios entre el nombre de la función, el primer paréntesis y el primer parámetro. Para el resto de los parámetros se insertará un espacio entre las comas y los parámetros. Tampoco existirán espacios entre el último parámetro, el paréntesis de cierre y el punto y coma (;).

```
$variable = funcion($parametro1, $parametro2 , $parametro3);
```

En caso de varias asignaciones, se asignarán espacios extras para aumentar la legibilidad.

```
$nombre_corto      = funcion($parametro);
```

```
$nombre_mucho_mas_largo = funcion($parametro);
```

- Declaración de funciones: Los parámetros con valores por defecto u opcionales irán al final de la lista de parámetros. Siempre se debe tratar de retornar una variable con un nombre significativo para la función.
- Arreglos: Los elementos y los operadores de asignación estarán separados por un espacio. En caso de que la sentencia sea mayor de los 80 caracteres o se realicen varias asignaciones en la sentencia, cada asignación se representará en una línea con un nivel de indentación. Puede observarse el siguiente fragmento de código de un módulo de Drupal:

```
$form['title'] = array(  
    '#type' => 'textfield',  
    '#title' => t('Title'),  
    '#size' => 60,  
    '#maxlength' => 128,  
    '#description' => t('Título de la página.'),  
);
```

- Concatenación de cadenas: Mantener un espacio a ambos lados de los operadores de concatenación (.) y (.=).
- Comentarios: Pueden utilizarse los símbolos (*//*) y (*/* ... */*) indistintamente. Los comentarios deben terminar con la puntuación adecuada.

- Nombres de funciones y miembros de clases: Las funciones deben nombrarse con letras minúsculas y las palabras separadas por un guión bajo (_). Los nombres de los miembros privados de las clases deben comenzar con un guión bajo.
- Constantes: Las constantes serán nombradas completamente con letras mayúsculas y las partes del nombre serán separadas por un guión bajo.

2.6 Conclusiones

En este capítulo se describieron los procesos del negocio referentes al trabajo con las incidencias en la UCI. Esto permitió un mejor entendimiento de dichos procesos, permitió identificar las dificultades presentes en su desarrollo y definir las funcionalidades con las que contará el sistema.

CAPÍTULO 3. PRUEBAS DEL SISTEMA

3.1 Introducción

En este capítulo se representan los casos de prueba necesarios para manifestar el posible cumplimiento de los requisitos mediante las pruebas de calidad, las cuales brindan la posibilidad de comprobar si la aplicación satisface los deseos del cliente, lo cual es el objetivo principal que se persigue con la realización de este trabajo.

3.2 Modelo de Pruebas

Las pruebas de calidad se basan en medir o evaluar lo que se espera de un módulo o sistema en general. Es decir, su principal objetivo es velar porque se cumplan los requerimientos definidos para el sistema. Las pruebas se denominan pruebas de caja blanca y pruebas de caja negra. Para comprobar la calidad de este producto se utilizará el tipo de prueba de caja negra y las mismas se realizarán por historias de usuario.

3.2.1 Casos de Prueba por historias de usuario

Tabla 3.1 Caso de prueba Autenticarse correctamente.

Caso de Prueba	
Nombre: Autenticarse correctamente	Nombre de la historia: Autenticar Usuarios
Descripción: Un usuario se conecta al sistema e intenta autenticarse insertando los datos correctamente.	
Condiciones de ejecución: El sistema debe tener habilitado el modulo LDAP_ UCI. El usuario debe tener asignado el rol Especialista de Seguridad y Protección o de Oficial Operativo.	
Entrada: El usuario llena los campos "Nombre de usuario" y "Contraseña" y pulsa el botón "Iniciar sesión".	
Resultados: El usuario entra al sistema, se le muestran únicamente las opciones a las que tiene permiso según su rol.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.2 Caso de prueba Autenticarse incorrectamente.

Caso de Prueba	
Nombre: Autenticarse incorrectamente.	Nombre de la historia: Autenticar Usuarios
Descripción: El Especialista de Seguridad y Protección intenta autenticarse insertando su contraseña incorrectamente.	
Condiciones de ejecución: El sistema debe tener habilitado el modulo LDAP_ UCI.	
Entrada: El Especialista de Seguridad y Protección llena los campos "Nombre de usuario" y "Contraseña" y pulsa el botón "Iniciar sesión".	
Resultados: El sistema muestra el mensaje: "Contraseña incorrecta".	
Evaluación de la prueba: Satisfactoria	

Tabla 3.3 Caso de prueba Control de acceso de usuarios no autorizados.

Caso de Prueba	
Nombre: Control de acceso de usuarios no autorizados.	Nombre de la historia: Autenticar Usuarios
Descripción: Un usuario que no está autorizado a trabajar en el sistema intenta autenticarse insertando su nombre de usuario y contraseña.	
Condiciones de ejecución: El sistema debe tener habilitado el modulo LDAP_ UCI.	
Entrada: El usuario llena los campos "Nombre de usuario" y "Contraseña" y pulsa el botón "Iniciar sesión".	
Resultados: El sistema muestra el mensaje:"El nombre de usuario (Nombre de usuario) no tiene permiso para acceder al sistema.".	
Evaluación de la prueba: Satisfactoria	

Tabla 3.4 Caso de prueba Control de acceso a la información.

Caso de Prueba	
Nombre: Control de acceso a la información.	Nombre de la historia: Administrar Usuarios
Descripción: El Oficial Operativo después de entrar al sistema selecciona la opción "Buscar Incidencias". Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. El Oficial Operativo selecciona el nombre de una de las incidencias en el campo "Asunto" de la tabla.	

Condiciones de ejecución: Debe existir al menos una incidencia registrada en el sistema.
Entrada: El Oficial Operativo selecciona el nombre de una incidencia.
Resultados: El sistema muestra el mensaje: “ Acceso Denegado No tiene autorización para acceder a esta página.”
Evaluación de la prueba: Satisfactoria

Tabla 3.5 Caso de prueba Mostrar el contenido de una Incidencia.

Caso de Prueba	
Nombre: Mostrar el contenido de una Incidencia.	Nombre de la historia: Mostrar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Incidencias”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de una de las incidencias en el campo “Imprimir” de la tabla.	
Condiciones de ejecución: Debe existir al menos una incidencia registrada en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de una incidencia.	
Resultados: El sistema muestra un documento en formato PDF correspondiente a la incidencia seleccionada.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.6 Caso de prueba Mostrar el contenido de un Expediente.

Caso de Prueba	
Nombre: Mostrar el contenido de un Expediente.	Nombre de la historia: Mostrar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes elaborados y enlaces a la información de los mismos. El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de uno de los expedientes en el campo “Imprimir” de la tabla.	
Condiciones de ejecución: Debe existir al menos un expediente almacenado en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de un expediente.	
Resultados: El sistema muestra un documento en formato PDF correspondiente al expediente seleccionado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.7 Caso de prueba Mostrar el contenido de un Infractor.

Caso de Prueba	
Nombre: Mostrar el contenido de un Infractor.	Nombre de la historia: Mostrar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Infractor”. Posteriormente aparece una página que contiene filtros de búsqueda y una tabla en la que se muestran datos de los infractores implicados en las diferentes incidencias reportadas y enlaces a la información de ellos. El Especialista de Seguridad y Protección selecciona la opción “Ver historial” de uno de los infractores en el campo “Historial” de la tabla.	
Condiciones de ejecución: Debe existir al menos el historial de un infractor almacenado en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ver historial” de un infractor.	
Resultados: El sistema muestra un documento en formato PDF correspondiente al historial del infractor seleccionado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.8 Caso de prueba Mostrar el contenido de un Informe Mensual.

Caso de Prueba	
Nombre: Mostrar el contenido de un Informe Mensual.	Nombre de la historia: Mostrar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Informe Mensual”. Posteriormente aparece una página que contiene un filtro de búsqueda y una tabla en la que se muestran datos de los informes elaborados y enlaces a la información de los mismos. El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de uno de los informes en el campo “Imprimir” de la tabla.	
Condiciones de ejecución: Debe existir al menos un informe almacenado en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de un informe.	
Resultados: El sistema muestra un documento en formato PDF correspondiente al informe seleccionado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.9 Caso de prueba Mostrar el contenido de un Reporte Semanal.

Caso de Prueba	
Nombre: Mostrar el contenido de un Reporte Semanal.	Nombre de la historia: Mostrar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Reporte Semanal”. Posteriormente aparece una página que muestra un filtro de búsqueda y una tabla en la que se presentan datos de los reportes elaborados y enlaces a la información de ellos. El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de uno de los reportes en el campo “Imprimir” de la tabla.	
Condiciones de ejecución: Debe existir al menos un reporte almacenado en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ver PDF” de un reporte.	
Resultados: El sistema muestra el documento en formato PDF correspondiente al reporte seleccionado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.10 Caso de prueba Validación de los campos numéricos del formulario Reportar Incidencia.

Caso de Prueba	
Nombre: Comprobar la validación de los campos numéricos del formulario Reportar Incidencia.	Nombre de la historia: Reportar una Incidencia
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Reportar Incidencia”. Posteriormente aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para reportar una incidencia. En el campo “Carnet de identidad”, introduce letras.	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección inserta letras en el campo “Carnet de identidad”.	
Resultados: El sistema muestra el mensaje: “Solo se aceptan valores numéricos.”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.11 Caso de prueba Validación de los campos obligatorios del formulario Reportar Incidencia.

Caso de Prueba	
Nombre: Comprobar la validación de los campos obligatorios del formulario Reportar Incidencia.	Nombre de la historia: Reportar una Incidencia
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Reportar Incidencia”. Posteriormente aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para reportar una incidencia, indicando con un asterisco rojo los campos que tienen carácter obligatorio. El Especialista pulsa el botón “Enviar”.	

Condiciones de ejecución:
Entrada: El Especialista de Seguridad y Protección deja los campos vacíos.
<p>Resultados: El sistema muestra un mensaje de error:</p> <p>“El campo Asunto es necesario. El campo Recibe el reporte es necesario. El campo Descripción es necesario. El campo Nombre es necesario. El campo Primer apellido es necesario. El campo Nombre es necesario. El campo Primer Apellido es necesario. El campo Carnet de identidad es necesario. El campo Solapín es necesario. “</p>
Evaluación de la prueba: Satisfactoria

Tabla 3.12 Caso de prueba Reportar una Incidencia.

Caso de Prueba	
Nombre: Reportar una Incidencia.	Nombre de la historia: Reportar una Incidencia
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Reportar Incidencia”. Posteriormente aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para reportar una incidencia. Inserta los datos necesarios y pulsa el botón “Enviar”.	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección llena los campos: asunto, recibe el reporte, fecha en que se detectó la incidencia, descripción, nombre y primer apellido del detector, nombre, primer apellido, carnet de identidad y solapín del informante y fecha y hora en que se informó la incidencia. Existen otros campos que son opcionales.	
Resultados: El sistema muestra el mensaje: “Se ha creado su <i>Incidencia.</i> ”, y debajo todos los datos de la incidencia creada.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.13 Caso de prueba Modificar datos de una Incidencia.

Caso de Prueba	
Nombre: Modificar datos de una Incidencia.	Nombre de la historia: Modificar una Incidencia
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Incidencias”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. Del campo “Modificar” de la tabla, selecciona la opción “Editar” de una de las incidencias. Seguidamente se muestra una página con el formulario que contiene los datos de la incidencia cuando se reportó. Modifica los datos que parecen en la sección “Datos del detector” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Debe existir al menos una incidencia registrada en el sistema.	
Entrada: El Especialista de Seguridad y Protección insertar datos nuevos en los campos “Nombre” y “Primer apellido” del detector.	
Resultados: El sistema actualiza los datos de la incidencia y muestra el mensaje: “Se ha actualizado la <i>Incidencia</i> .”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.14 Caso de prueba Eliminar una Incidencia.

Caso de Prueba	
Nombre: Eliminar una Incidencia.	Nombre de la historia: Eliminar una Incidencia
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Incidencias”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. Del campo “Eliminar” de la tabla, selecciona la opción “Borrar” de una de las incidencias. El sistema muestra un mensaje: “¿Está seguro que quiere borrar (nombre de la incidencia)? No se puede deshacer esta acción”. El Especialista pulsa el botón “Borrar”.	
Condiciones de ejecución: Debe existir al menos una incidencia registrada en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Borrar” de una incidencia y pulsa el botón “Borrar” que aparece en el mensaje de confirmación.	
Resultados: El sistema elimina la incidencia seleccionada y muestra el mensaje: “Se ha eliminado (nombre de la incidencia).”	
Evaluación de la prueba: Satisfactoria	

Tabla 3.15 Caso de prueba Validación de los campos numéricos del formulario Crear Expediente.

Caso de Prueba	
Nombre: Comprobar validación de los campos numéricos para crear un expediente.	Nombre de la historia: Crear un Expediente
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Crear Expediente”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. Del campo “Expediente” de la tabla selecciona la opción “Expediente”, de una incidencia a la que no se le haya creado ningún expediente. Seguidamente se mostrará una página que contiene un formulario en el que aparecen los campos que deben llenarse para elaborar el expediente. En el campo “Número del expediente”, introduce letras.	
Condiciones de ejecución: Debe existir al menos una incidencia en el sistema que no tenga ningún expediente asociado.	
Entrada: El Especialista de Seguridad y Protección inserta letra en el campo “Número del Expediente”.	
Resultados: El sistema muestra el mensaje: ” Solo se aceptan valores numéricos.”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.16 Caso de prueba Validación de los campos obligatorios del formulario Crear Expediente.

Caso de Prueba	
Nombre: Comprobar la validación de los campos obligatorios del formulario Crear Expediente.	Nombre de la historia: Crear un Expediente
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Crear Expediente”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. Del campo “Expediente” de la tabla selecciona la opción “Expediente”, de una incidencia a la que no se le haya creado ningún expediente. Seguidamente aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para elaborar el expediente, indicando con un asterisco rojo los campos que tienen carácter obligatorio. El Especialista pulsa el botón “Enviar”.	
Condiciones de ejecución: Debe existir al menos una incidencia en el sistema que no tenga ningún expediente asociado.	
Entrada: El Especialista de Seguridad y Protección deja los campos vacíos.	

<p>Resultados: El sistema muestra el mensaje:</p> <p>“El campo Elaborado por es necesario. El campo Número del expediente es necesario. El campo Lugar del hecho es necesario. El campo Descripción es necesario. El campo Medios afectados es necesario.”</p>
<p>Evaluación de la prueba: Satisfactoria</p>

Tabla 3.17 Caso de prueba Crear un Expediente.

Caso de Prueba	
Nombre: Crear un Expediente.	Nombre de la historia: Crear un Expediente
<p>Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Crear Expediente”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de las incidencias reportadas y enlaces a la información de las mismas. Del campo “Expediente” de la tabla selecciona la opción “Expediente”, de una incidencia a la que no se le haya creado ningún expediente. Aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para elaborar el expediente. Inserta los datos necesarios y pulsa el botón “Enviar”.</p>	
<p>Condiciones de ejecución: Debe existir al menos una incidencia en el sistema que no tenga ningún expediente asociado.</p>	
<p>Entrada: El Especialista de Seguridad y Protección llena los campos: elaborado por, tipificación, número del expediente, fecha y hora de elaboración, estado, lugar del hecho, descripción, medios afectados, protección del lugar, esclarecimiento y clasificación. Existen otros campos que son opcionales.</p>	
<p>Resultados: El sistema muestra el mensaje: “Se ha creado su <i>Expediente</i>.”, y debajo todos los datos del expediente creado.</p>	
<p>Evaluación de la prueba: Satisfactoria</p>	

Tabla 3.18 Caso de prueba Modificar un Expediente.

Caso de Prueba	
Nombre: Modificar datos de un expediente.	Nombre de la historia: Modificar un Expediente
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes creados y enlaces a la información de los mismos. Del campo “Modificar” de la tabla, selecciona la opción “Editar” de cualquiera de los expedientes que aparecen en la misma. Seguidamente se muestra una página que contiene un formulario en el que aparecen los campos que deben llenarse para elaborar un expediente. Modifica los datos que aparecen en la sección “Detalles” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Debe existir al menos un expediente en el sistema.	
Entrada: El Especialista de Seguridad y Protección llena los campos “Lugar del hecho” y “Descripción”.	
Resultados: El sistema actualiza los datos del expediente y muestra el mensaje: “Se ha actualizado el <i>Expediente</i> .”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.19 Caso de prueba Eliminar un Expediente.

Caso de Prueba	
Nombre: Eliminar un Expediente.	Nombre de la historia: Eliminar un Expediente
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes creados y enlaces a la información de los mismos. Del campo “Eliminar” de la tabla, selecciona la opción “Borrar” de cualquiera de los expedientes que aparecen en la tabla. El sistema muestra un mensaje: “¿Está seguro que quiere borrar (nombre del expediente)? No se puede deshacer esta acción”. El Especialista pulsa el botón “Borrar”.	
Condiciones de ejecución: Debe existir al menos un expediente en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Borrar” de un expediente y pulsa el botón “Borrar” que aparece en el mensaje de confirmación.	
Resultados: El sistema elimina el expediente y muestra el mensaje: “Se ha eliminado (nombre del expediente).”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.20 Caso de prueba Adicionar un Infractor.

Caso de Prueba	
Nombre: Adicionar un Infractor.	Nombre de la historia: Adicionar un Infractor
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes creados y enlaces a la información de los mismos. Del campo “Modificar” de la tabla, selecciona la opción “Editar” de cualquiera de los expedientes que aparecen en la misma. Seguidamente se muestra una página que contiene un formulario en el que aparecen los campos que deben llenarse para elaborar un expediente. De la sección “Infractores” selecciona la opción “Adicionar infractor desconocido”. A continuación aparece una página nueva con un formulario en el que se solicitan los datos necesarios para adicionar un infractor. Inserta los datos necesarios y pulsa el botón “Enviar”.	
Condiciones de ejecución: Debe existir al menos un expediente en el sistema.	
Entrada: El Especialista de Seguridad y Protección llena los campos “Nombre(s) y apellidos”, “Carnet de identidad” y “Procedencia”. Existen otros campos que son opcionales.	
Resultados: El sistema muestra el mensaje: “Se ha creado su <i>Infractor.</i> ” y debajo todos los datos del infractor creado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.21 Caso de prueba Validación de los campos obligatorios del formulario Adicionar Infractor.

Caso de Prueba	
Nombre: Comprobar la validación de los campos obligatorios del formulario Adicionar Infractor.	Nombre de la historia: Adicionar un Infractor
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes creados y enlaces a la información de los mismos. Del campo “Modificar” de la tabla, selecciona la opción “Editar” de cualquiera de los expedientes que aparecen en la misma. A continuación aparece una página que contiene un formulario en el que muestra los datos que se insertaron cuando se creó el expediente. De la sección “Infractores” selecciona la opción “Adicionar infractor desconocido”. Seguidamente se muestra una página nueva que contiene un formulario en el que aparecen los campos que deben ser llenados para adicionar un infractor, indicando con un asterisco rojo los campos que tienen carácter obligatorio. El Especialista pulsa el botón “Enviar”.	
Condiciones de ejecución: Debe existir al menos un expediente en el sistema.	

Entrada: El Especialista de Seguridad y Protección deja los campos vacíos.
Resultados: El sistema muestra el mensaje: “El campo Nombre(s) y apellidos es necesario. El campo Carnet de identidad es necesario.”
Evaluación de la prueba: Satisfactoria

Tabla 3.22 Caso de prueba Validación de los campos numéricos del formulario Adicionar Infractor.

Caso de Prueba	
Nombre: Comprobar la validación de los campos numéricos del formulario Adicionar Infractor.	Nombre de la historia: Adicionar un Infractor
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los expedientes creados y enlaces a la información de los mismos. Del campo “Modificar” de la tabla, selecciona la opción “Editar” de cualquiera de los expedientes que aparecen en la misma. Seguidamente aparece una página que muestra un formulario que contiene los datos que se insertaron cuando se creó el expediente. De la sección “Infractores” selecciona la opción “Adicionar infractor desconocido”. A continuación aparece una página nueva con un formulario en el que se solicitan los datos necesarios para adicionar un infractor. El Especialista inserta letras en el campo “Carnet de identidad”.	
Condiciones de ejecución: Debe existir al menos un expediente en el sistema.	
Entrada: El Especialista de Seguridad y Protección inserta letras en el campo “Carnet de identidad”.	
Resultados: El sistema muestra el mensaje: ” Solo se aceptan valores numéricos.”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.23 Caso de prueba Modificar los datos de un Infractor.

Caso de Prueba	
Nombre: Modificar los datos de un Infractor.	Nombre de la historia: Modificar un Infractor
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Infractor”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los infractores vinculados a las diferentes incidencias y enlaces a la información de los mismos. Selecciona la opción “Editar” de cualquiera de los infractores que aparecen en la tabla. Seguidamente se muestra una página que contiene un formulario en el que aparecen los datos que se insertaron cuando se creó el infractor. Modifica los datos de los campos “Solapín” y “Carnet de identidad” y pulsa el botón “Enviar”.	

Condiciones de ejecución: Debe existir al menos un infractor en el sistema.
Entrada: El Especialista de Seguridad y Protección inserta los datos nuevos en los campos “Solapín” y “Carnet de identidad”.
Resultados: El sistema actualiza los datos del infractor y muestra el mensaje:”Se ha actualizado el <i>Infractor</i> .”.
Evaluación de la prueba: Satisfactoria

Tabla 3.24 Caso de prueba Eliminar un Infractor.

Caso de Prueba	
Nombre: Eliminar un Infractor.	Nombre de la historia: Eliminar un Infractor
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Infractor”. Posteriormente aparece una página que muestra filtros de búsqueda y una tabla en la que se presentan datos de los infractores vinculados a las diferentes incidencias y enlaces a la información de los mismos. Selecciona la opción “Borrar” de cualquiera de los infractores que aparecen en la tabla. El sistema muestra un mensaje: “¿Está seguro que quiere borrar (nombre del infractor)? No se puede deshacer esta acción”. El Especialista pulsa el botón “Borrar”.	
Condiciones de ejecución: Debe existir al menos un infractor en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Borrar” de un infractor y pulsa el botón “Borrar” del mensaje de confirmación.	
Resultados: El sistema elimina el infractor y muestra el mensaje:”Se ha eliminado (Nombre del infractor).”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.25 Caso de prueba Validación de los campos obligatorios del formulario Crear Reporte Semanal.

Caso de Prueba	
Nombre: Comprobar la validación de los campos obligatorios del formulario Crear Reporte Semanal.	Nombre de la historia: Crear un Reporte Semanal
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Crear Reporte Semanal”. Posteriormente aparece una página que muestra un formulario en el que aparecen los campos que deben llenarse para elaborar un reporte, indicando con un asterisco rojo los campos que tienen carácter obligatorio. El Especialista pulsa el botón “Enviar”.	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección deja los campos vacíos.	

Resultados: El sistema muestra el mensaje: "El campo Semana es necesario."
Evaluación de la prueba: Satisfactoria

Tabla 3.26 Caso de prueba Crear un Reporte Semanal.

Caso de Prueba	
Nombre: Crear un Reporte Semanal.	Nombre de la historia: Crear un Reporte Semanal
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción "Crear Reporte Semanal". Posteriormente aparece una página que muestra un formulario en el que se aparecen los campos que deben llenarse para crear un reporte. El Especialista inserta los datos necesarios y pulsa el botón "Enviar".	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección llena los campos: semana, fecha de inicio y fecha final. Existen otros campos que son opcionales.	
Resultados: El sistema muestra un mensaje: "Se ha creado su <i>Reporte Semanal.</i> " y debajo todos los datos del reporte creado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.27 Caso de prueba Modificar datos de un Reporte Semanal.

Caso de Prueba	
Nombre: Modificar datos de un Reporte Semanal.	Nombre de la historia: Modificar un Reporte Semanal
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción "Buscar Reporte Semanal". Posteriormente aparece una página que muestra un filtro de búsqueda y una tabla en la que se presentan datos de los reportes creados y enlaces a la información de los mismos. Selecciona la opción "Editar" de uno de los reportes que aparecen en la tabla. Seguidamente aparece una página que contiene un formulario en el que se muestran los datos que se insertaron cuando se creó el reporte. El Especialista modifica los datos de los campos "Fecha de inicio" y "Fecha final" y pulsa el botón "Enviar".	
Condiciones de ejecución: Tiene que existir al menos un reporte en el sistema.	
Entrada: El Especialista de Seguridad y Protección inserta los datos nuevos en los campos "Fecha de inicio" y "Fecha final".	

Resultados: El sistema actualiza los datos del reporte y muestra el mensaje: "Se ha actualizado el <i>Reporte Semanal</i> ".
Evaluación de la prueba: Satisfactoria

Tabla 3.28 Caso de prueba Eliminar un Reporte Semanal.

Caso de Prueba	
Nombre: Eliminar un Reporte Semanal.	Nombre de la historia: Eliminar un Reporte Semanal
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción "Buscar Reporte Semanal". Posteriormente aparece una página que muestra un filtro de búsqueda y una tabla en la que se presentan datos de los reportes creados y enlaces a la información de los mismos. Selecciona la opción "Borrar" de uno de los reportes que aparecen en la tabla. El sistema muestra un mensaje: "¿Está seguro que quiere borrar (nombre del reporte)? No se puede deshacer esta acción". El Especialista pulsa el botón "Borrar".	
Condiciones de ejecución: Tiene que existir al menos un reporte en el sistema.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción "Borrar" de un reporte y pulsa el botón "Borrar" que aparece en el mensaje de confirmación.	
Resultados: El sistema elimina el reporte y muestra el mensaje: "Se ha eliminado (Nombre del reporte.)".	
Evaluación de la prueba: Satisfactoria	

Tabla 3.29 Caso de prueba Validación de los campos obligatorios del formulario Crear un Informe Mensual.

Caso de Prueba	
Nombre: Comprobar la validación de los campos obligatorios del formulario Crear un Informe Mensual.	Nombre de la historia: Crear un Informe Mensual
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción "Crear Informe Mensual". Posteriormente aparece una página que contiene un formulario en el que aparecen los campos que deben llenarse para crear un informe, indicando con un asterisco rojo los campos que tienen carácter obligatorio. El Especialista pulsa el botón "Enviar".	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección deja los campos en blanco.	
Resultados: El sistema muestra el mensaje: "El campo encabezado del informe es necesario."	
Evaluación de la prueba: Satisfactoria	

Tabla 3.30 Caso de prueba Crear un Informe Mensual.

Caso de Prueba	
Nombre: Crear un Informe Mensual.	Nombre de la historia: Crear un Informe Mensual
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Crear Informe Mensual”. Posteriormente aparece una página que contiene un formulario en el que se muestran los campos que deben llenarse para crear un informe mensual. El Especialista de Seguridad y Protección inserta los datos necesarios y pulsa el botón “Enviar”.	
Condiciones de ejecución:	
Entrada: El Especialista de Seguridad y Protección llena los campos: encabezado del informe, fecha inicial, fecha final y mes a informar. Existen otros campos que son opcionales.	
Resultados: El sistema muestra un mensaje de error:”Se ha creado su <i>Informe Mensual</i> .” y debajo los datos del informe creado.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.31 Caso de prueba Modificar datos de un Informe Mensual.

Caso de Prueba	
Nombre: Modificar datos de un Informe Mensual.	Nombre de la historia: Modificar un Informe Mensual
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Informe Mensual”. Posteriormente aparece una página que muestra un filtro de búsqueda y una tabla en la que se presentan datos de los informes creados y enlaces a la información de los mismos. Selecciona la opción “Editar” de cualquiera de los informes que aparecen en la tabla. A continuación aparece una página que contiene un formulario en el que se muestran los datos que se insertaron cuando se creó el reporte. Modifica los datos del campo “Fecha inicial” y “Fecha final” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos un informe almacenado en el sistema.	
Entrada: El Especialista de Seguridad y Protección inserta los datos nuevos en los campos “Fecha inicial” y “Fecha final”.	
Resultados: El sistema actualiza los datos del informe y muestra el mensaje:”Se ha actualizado el <i>Informe Mensual</i> .”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.32 Caso de prueba Eliminar un Informe Mensual.

Caso de Prueba	
Nombre: Eliminar un Informe Mensual.	Nombre de la historia: Eliminar un Informe Mensual
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Informe Mensual”. Posteriormente aparece una página que muestra un filtro de búsqueda y una tabla en la que se presentan datos de los informes creados y enlaces a la información de los mismos. Selecciona la opción “Borrar” de cualquiera de los informes creados. El sistema muestra un mensaje: “¿Está seguro que quiere borrar (nombre del informe? No se puede deshacer esta acción”. El Especialista pulsa el botón “Borrar”.	
Condiciones de ejecución: Tiene que existir al menos un informe en el sistema.	
Entrada: El Especialista de Seguridad y Protección Selecciona la opción “Borrar” de un informe y pulsa el botón “Borrar” que aparece en el mensaje de confirmación.	
Resultados: El sistema elimina el informe y muestra el mensaje:”Se ha eliminado (Nombre del Informe.)”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.33 Caso de prueba Buscar una Incidencia.

Caso de Prueba	
Nombre: Buscar una Incidencia.	Nombre de la historia: Buscar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Incidencias”. Posteriormente aparece una página que muestra filtros de búsquedas y una tabla en que se presentan datos de las incidencias reportadas hasta el momento. Del campo “Mes” selecciona la opción “Ene” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos una incidencia en el sistema que haya sido reportada en el mes de enero.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ene”.	
Resultados: El sistema muestra una tabla con los datos de las incidencias reportadas en el mes de enero.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.34 Caso de prueba Buscar un Expediente.

Caso de Prueba	
Nombre: Buscar un Expediente.	Nombre de la historia: Buscar contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Expedientes”. Posteriormente aparece una página que muestra filtros de búsquedas y una tabla en que se presentan datos de los expedientes elaborados y enlaces a los mismos. Del campo “Mes” selecciona la opción “Ene” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos un expediente en el sistema que haya sido elaborado en el mes de enero.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Ene”.	
Resultados: El sistema muestra una tabla con los datos de los expedientes elaborados en el mes de enero.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.35 Caso de prueba Buscar un Infractor.

Caso de Prueba	
Nombre: Buscar un Infractor.	Nombre de la historia: Buscar Contenido
Descripción: El Especialista de Seguridad y Protección después de entra al sistema selecciona la opción “Buscar Infractor”. Posteriormente aparece una página que muestra filtros de búsquedas y una tabla en que se presentan datos de los infractores almacenados en sistema y enlaces a la información de ellos. En el campo “Ocupación” inserta “Técnico de laboratorio” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos un infractor cuya ocupación sea “Técnico de laboratorio”.	
Entrada: El Especialista de Seguridad y Protección inserta “Técnico de laboratorio” en el campo “Ocupación”.	
Resultados: El sistema muestra una tabla con los datos del los infractores cuya ubicación es “Técnico de laboratorio”.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.36 Caso de prueba Buscar un Reporte Semanal.

Caso de Prueba	
Nombre: Buscar un Reporte Semanal.	Nombre de la historia: Buscar Contenido
Descripción: El Especialista de Seguridad y Protección después de entrar al sistema selecciona la opción “Buscar Reporte Semanal”. Posteriormente aparece una página que muestra un filtro de búsquedas y una tabla en la que se presentan datos de los reportes creados y enlaces a la información de los mismos. En el campo “Semana” inserta el nombre de la semana de los reportes que desea buscar y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos un reporte en el sistema que haya sido elaborado en esa fecha.	
Entrada: El Especialista de Seguridad y Protección inserta “2 de noviembre al 9 de noviembre” en el campo “Semana”.	
Resultados: El sistema muestra una tabla con los reportes creados en ese período de tiempo.	
Evaluación de la prueba: Satisfactoria	

Tabla 3.37 Caso de prueba Buscar Contenido.

Caso de Prueba	
Nombre:	Nombre de la historia: Buscar Contenido
Descripción: El Especialista de Seguridad después de entrar al sistema selecciona la opción “Buscar Informe Mensual”. Posteriormente aparece una página que muestra un filtro de búsquedas y una tabla en la que se presentan datos de los reportes creados y enlaces a la información de los mismos. En el campo “Mes a informar” selecciona la opción “Noviembre” y pulsa el botón “Enviar”.	
Condiciones de ejecución: Tiene que existir al menos un informe en el sistema que haya sido elaborado en ese mes.	
Entrada: El Especialista de Seguridad y Protección selecciona la opción “Noviembre”.	
Resultados: El sistema muestra una tabla con los datos de los informes creados en ese mes.	
Evaluación de la prueba: Satisfactoria	

3.3 Conclusiones

Los casos de prueba representados en este capítulo permitieron comprobar que las funcionalidades del sistema están bien realizadas, debido a que los resultados de todas las pruebas fueron satisfactorios, por lo que se puede concluir diciendo que el sistema cumple con el objetivo para el cual fue diseñado.

CONCLUSIONES

En el presente trabajo se le da solución informática a la situación problemática planteada. Para ello se realizó el diseño e implementación de un sistema que permite disminuir considerablemente las dificultades existentes con la gestión de expedientes de incidencias en la Oficina de Seguridad y Protección y llevar un control más eficiente sobre los incidentes ocurridos en la universidad.

A partir de entrevistas realizadas en la Oficina de Seguridad y Protección y en el Puesto de Mando UCI, se conocieron las necesidades de la institución en cuanto a la realización de los procesos de gestión de incidencias y sobre la base de esta investigación se diseñó la aplicación.

Se concluye además que la utilización del sistema en la Oficina de Seguridad de Seguridad y Protección posibilita la seguridad de la información con la que se trabaja ya que la misma se encontrará digitalizada en la base de datos del sistema, facilita las búsquedas de los datos en un corto tiempo y sin errores y brinda la posibilidad de disminuir los gastos económicos producto del consumo de material de oficina.

Con la elaboración del presente trabajo se logró consolidar los conocimientos adquiridos a lo largo de la carrera y obtener otros que fueron necesarios para el desarrollo del mismo.

RECOMENDACIONES

Añadirle funcionalidades al sistema como:

- Importar datos en formato Excel
- Almacenar y consultar documentos Word y PDF.
- Integrar las funcionalidades anteriores mediante el envío y recepción de correo electrónico.

Dichas funcionalidades permitirán recibir información ya procesada e incluirla correctamente al sistema, sin necesidad de teclearla nuevamente.

REFERENCIAS BIBLIOGRÁFICAS

- [1]. INFORMATION MANAGEMENT. ¿Qué es la gestión de la información? [En línea] 2006. [Citado el: 19 de Septiembre de 2008.]
<http://informationmanagement.wordpress.com/2006/11/28/%c2%bfque-es-la-gestion-de-la-informacion-3-de-4/>
- [2]. POLIKS. Information, communication and administration - a networked process platform for police forces. [En línea] 2005. [Citado el: 20 de Septiembre de 2008.]
http://download.sczm.t-systems.de/t-systems.de/en/StaticPage/39/05/18/390518_Download-ps.pdf
- [3]. Secretaría General de INTERPOL. INTERPOL EN ACCIÓN. [En línea] 2005. [Citado el: 22 de Septiembre de 2008.] <http://www.interpol.int/Public/ICPO/InterpolAtWork/iaw2005es.pdf>
- [4]. Lic. Analía Jaime, Amalia Smaha, Dr. Néstor Pares, Maximiliano Jaime. *Plan de Identificación Biométrica de la Provincia de Mendoza*. 2006.
- [5]. Rejero, José A. Sobre Drupal. [En línea] 16 de Marzo de 2006. [Citado el: 13 de Octubre de 2008.]
<http://drupal.org.es/drupal>
- [6]. Martínez, Carlos Casado. Páginas Web estáticas vs páginas Web dinámicas. Las bases de datos en Internet. [En línea] 29 de Marzo de 2004. [Citado el: 22 de Septiembre de 2008.]
http://mosaic.uoc.edu/old/2_opinion2/modeljuny/carlos_casado.html
- [7]. Castillo, Alvaro del. Webs dinámicos con PHP. [En línea] 2006. [Citado el: 24 de Septiembre de 2008.]
<http://www.programacion.net/php/tutorial/php4/2/>
- [8]. Sistemas gestores de bases de datos. [En línea] 31 de Julio de 2007. [Citado el: 11 de Octubre de 2008.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>
- [9]. PostgreSQL Global Development Group PostgreSQL. [En línea] [Citado el: 11 de Octubre de 2008.]
<http://www.postgresql.org>
- [10]. Pérez, José Manuel. ¿Qué es MySQL? [En línea] 16 de Agosto de 2005. [Citado el: 12 de Octubre de 2008.]
<http://www.esepestudio.com/articulo/desarrollo-web/bases-de-datos-mysql/Que-es-MySQL.htm>
- [11]. Romero, Diego F. ¿Qué Es Un Servidor De Aplicaciones? [En línea] 2007. [Citado el: 8 de Octubre de 2008.] <http://www.editum.org/Que-Es-Un-Servidor-De-Aplicaciones-p-473.html>

- [12]. Foundation, Apache Software. Apache. [En línea] [Citado el: 10 de Octubre de 2008.] <http://www.apache.org>
- [13] Kniberg, Henrik. *Scrum y XP desde las trincheras*. Canadá: s.n., 2007.p.119.
- [14]. Palacio, Juan. Flexibilidad con Scrum. 2007.
- [15]. Patricio Letelier, María Carmen Penadés. Metodologías ágiles para el desarrollo del software: Extreme Programming (XP). Valencia: s.n.

BIBLIOGRAFÍA

1. Apache Software Foundation. Apache. [En línea] [Citado el: 29 de Septiembre de 2008.] <http://www.apache.org>
2. Ariel Enrique Novo Rijo, Laritza González Marrero. Arquitectura del Sistema Informático de Cooperación Internacional. Ciudad de la Habana: s.n, 2008. P.69.
3. Desarrolloweb.com. Zend Studio. 2008 [En línea] [Citado el: 18 de Noviembre de 2008.] <http://www.desarrolloweb.com/articulos/1178.php>
4. Drupal Hispano. [En línea] [Citado el: 15 de Noviembre de 2008.] <http://drupal.org.es/>
5. Jacobson, Booch, Rumbaugh. *El Proceso Unificado de Desarrollo de Software*. Madrid, Pearson educación, S.A, 2000. P.464.
6. Kniberg, Henrik. *Scrum y XP desde las trincheras*. Canadá: s.n., 2007.p.119.
7. La Web del Programador. [En línea] [Citado el: 20 de Noviembre de 2008.] <http://www.lawebdelprogramador.com/>
8. MANUAL DE PHP. [En línea] [Citado el: 26 de Septiembre de 2008.] <http://www.manualdephp.es/>
9. Palacio, Juan. Flexibilidad con Scrum. 2007. P.175.
10. Patricio Letelier, María Carmen Penadés. *Metodologías ágiles para el desarrollo del software: Extreme Programming (XP)*. Valencia: s.n.P.16.
11. Paz, Jose Dapena. *Desarrollo en comunidad con Extreme Programming*. España: s.n.P.11.
12. Peralta, Adriana. *Metodología Scrum (Universidad ORT Uruguay, Facultad de Ingeniería)*. Montevideo: s.n., 2003. P.9.
13. Secretaría General de INTERPOL. INTERPOL EN ACCIÓN. [En línea] 2005. [Citado el: 22 de Septiembre de 2008.] <http://www.t-systems.com/>
14. T-System. [En línea] [Citado el: 20 de Septiembre de 2008.] <http://www.t-systems.com/>
15. Velázquez, Mairelys Boeras. Sistema de Gestión de Incidencias para la Universidad de las Ciencias Informáticas. Ciudad de la Habana: s.n., 2006. P.109.
16. Web oficial de PHP. [En línea] [Citado el: 24 de Septiembre de 2008.] <http://www.php.net/>

GLOSARIO DE TÉRMINOS

Proceso: En informática se manejan varias definiciones que aluden a diversos elementos: puede ser simplemente una operación o conjunto combinado de operaciones con datos, o bien una secuencia de acontecimientos definida única y delimitada, que obedece a una intención operacional en condiciones predeterminadas. También se denomina proceso a una función que se está ejecutando.

Tomado de: Glosario de Informática e Internet.

http://www.casarramona.com/mt/programador/diccionario/dic_p.html

JavaScript: Lenguaje desarrollado por Netscape y aunque es parecido a Java se diferencia de él en que los programas están incorporados en el archivo HTML.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-j.html>

NCSA: National Center for Supercomputing Applications (Centro Nacional de Aplicaciones de Supercomputación) es un organismo estadounidense relacionado con la investigación en el campo de la Informática y las Telecomunicaciones.

Tomado de: <http://www.glosarium.com/term/443,4,xhtml>

RUP: Rational Unified Process (Proceso Unificado de Desarrollo) es la metodología de IBM Rational para el desarrollo y construcción de software basado íntegramente en UML como soporte a la metodología.

Tomado de: Jacobson, Booch, Rumbaugh. *El Proceso Unificado de Desarrollo de Software*. Madrid, Pearson educación, S.A, 2000. P.464.

UML: Unified Modeling Language (Lenguaje Unificado de Modelado) El Lenguaje Unificado de Modelado 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.

Tomado de: Popkin Software and Systems, Modelado de Sistemas con UML. P.532.

HTTP: HyperText Transfer Protocol Protocolo de Transferencia de Hipertexto. HTTP es un protocolo con la ligereza y velocidad necesaria para distribuir y manejar sistemas de información hipermedia.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-h.html>

CMS: Content Management System (Sistema de gestión de contenidos) es un programa que se refiere usualmente a sitios web que tienen sistemas de administración que permiten la fácil creación y edición de contenidos, como páginas nuevas, noticias, etc.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-c.html>

LDAP: Lightweight Directory Access Protocol (Protocolo Ligero de Acceso a Directorios) es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-l.html>

XML: Extensible Markup Language (lenguaje de marcas) es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-x.html>

HTML: HyperText Markup Language (Lenguaje de Marcas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-h.html>

CSS: Cascading Style Sheets (hojas de estilo en cascada) Conjunto de instrucciones HTML que definen la apariencia de uno o más elementos de un conjunto de páginas web con el objetivo de uniformizar su diseño.

Tomado de: Glosario de Informática e Internet. <http://www.panamacom.com/glosario/letra-c.html>

IDE: Integrated Development Environment (Entorno de desarrollo integrado) es un programa compuesto por un conjunto de herramientas para un programador.

Tomado de: Babylon. http://www.babylon.com/definicion/entorno_de_desarrollo/Spanish

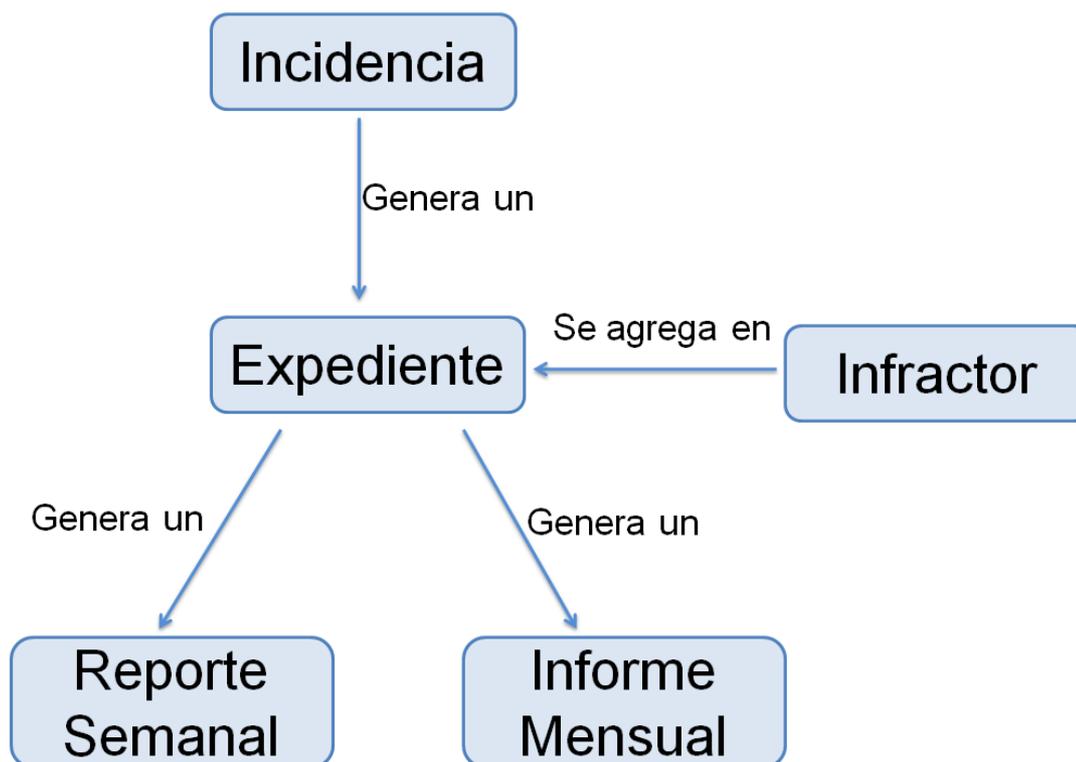
SGBD: DataBase Management System (Sistemas de Gestión de Base de Datos) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

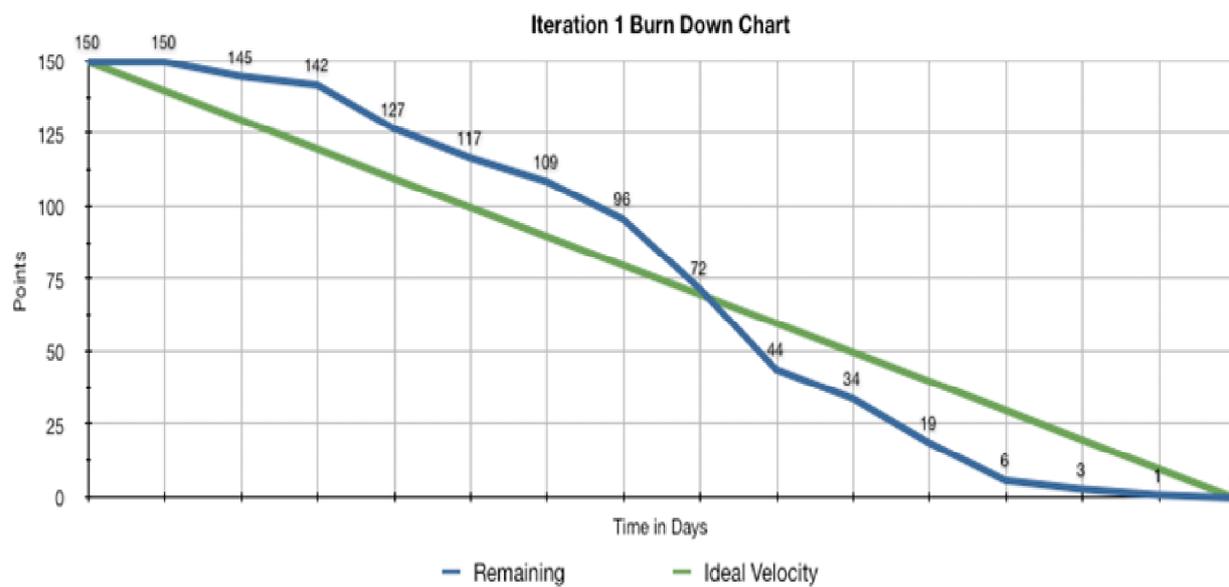
Tomado de: Babylon.

http://www.babylon.com/definicion/sistema_de_gesti%C3%B3n_de_base_de_datos/Spanish

ANEXOS

Anexo 1: Mapa Conceptual



Anexo 2: Diagrama Burndown chart

Anexo 3: Diagrama Burndown chart

