

Universidad de las Ciencias Informáticas

Facultad 2



Trabajo de diploma para optar por el título de Ingeniero en
Ciencias Informáticas.

**Herramienta informática para la creación y restauración de
copias de seguridad sobre el Gestor de Documentos
Administrativos Xabal eXscriba 3.1.**

Autores:

Glauver Matos Fonseca

Mirna Eulalia Despaigne Biset

Tutor:

Ing. Yaniel Blanco Fernández

La Habana, Cuba 2016

“Año 58 de la Revolución”

Declaración de autoría

Declaramos ser los autores del presente trabajo de diploma y autorizamos a la Facultad 2 de la Universidad de las Ciencias Informáticas; así como a dicho centro para que hagan el uso que estimen pertinente con carácter exclusivo de este trabajo.

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

Firma del autor

Mirna E. Despaigne Biset

Firma del autor

Glauver Matos Fonseca

Firma del tutor

Ing. Yaniel Blanco Fernández

Dedicatoria

Mirna

A madre, mi hermana y en general a toda mi familia por ser incondicionales. A mi abuela (que dios la tenga en la gloria) por ser siempre el motor impulsor, cuando en un momento de mi vida pensé que había perdido completamente la posibilidad de realizar uno de mis sueños "Ser ingeniera en Ciencias Informáticas".

A mi novio Alejandro por convertirse en una persona muy importante en mi vida, gracias por apoyarme y comprenderme durante este difícil año.

Glauver

A mis padres por ser siempre mi fuente de inspiración y los artífices de mi existencia.

Agradecimientos

Mirna E. Despaigne Biset

Primeramente, ofrezco mi más sincero agradecimiento a Dios por darme salud y permitirme llegar a este momento tan deseado por todos los estudiantes que transitan las calles de esta universidad. En segundo lugar, deseo felicitarme por mi dedicación, constancia y fundamentalmente mi perseverancia para lograr este sueño; que hoy se convierte en un hecho.

Al tribunal por su comprensión y por su colaboración para la realización de este trabajo. A mi tutor por su esfuerzo y paciencia. A mi compañero de tesis por brindarme la posibilidad de trabajar junto él. A todos los profesores que de una manera u otra aportaron su granito de arena para ayudar en la creación de la tesis, en especial a mi profesora guía Yeni y al profesor Carlos Luis Milián por apoyarme y ayudarme en las disciplinas de Inteligencia Artificial. En general agradecerle a todos los que influyeron en mi formación profesional.

A mi familia por siempre estar ahí a mi lado, a mi novio por todo el apoyo incondicional que me ha ofrecido en los momentos más tensos y frustrantes de mí vida. En el transcurso de la especialidad perdí a personas muy queridas que influyeron muchísimo en mi formación como persona. Gracias mi abuelita por ser siempre mi motor impulsor, aunque físicamente ya no estés entre nosotros. Cuantos recuerdos llegan a mi cabeza, malos y buenos, difíciles y no tan difíciles, grandes y pequeños; pero todos tienen dos factores humanos en común: tú y yo. Siempre estuviste ahí para darme un consejo lleno de sabiduría y experiencia. Te sigo queriendo igual o más que ayer.

A mis enemigos por hacerme más fuerte cada día y por enseñarme a mantenerme en pie ante la traición, la hipocresía, la envidia y el engaño. También agradecerles a mis compañeros de clases, los antiguos y los que no son tan antiguos: Jordan, Nesty, Yamila, Daynier, Héctor, Felipe, Paloma, Eyidey, Víctor, Eddry, Dariel (mi mantecadito), Reynier, Elvis... A las personas que me han demostrado con sus bellas acciones y sentimientos el significado de la palabra amistad: Saralys, Yanet, mi hermana Mailín, mis tías, especialmente mi tía Magalys y mi tía Delmis. Y no por último es menos importante, mi madre Una madre es siempre una madre. Cuantas noches de desvelo, preocupaciones y lágrimas derramadas. Te quiero mucho por esforzarte tanto para sacarnos adelante a mi hermana y a mí, y entregarnos toda una vida llena de amor y dedicación.

Glauver Matos Fonseca

Quiero agradecer a mi familia que de una forma u otra me ha apoyado durante el transcurso de la carrera. En especial agradecer a mi mamá que, a pesar de no poder estar presente hoy aquí, me ha brindado su apoyo desde el que día que me trajo al mundo. Ella es todo para mí. A mi papá, gracias a él, proponerse metas fue más fácil, contar con su guía y apoyo fue muy importante, tener su presencia hoy me hace muy feliz. A mi hermana que a pesar de no poder estar aquí también me dio ánimos y consejos, hermana te quiero mucho. A mi novia y futura esposa lleguen mis agradecimientos. Agradecer a mis abuelas, Nelly e Idelisa, a mis tías Teresa, Maricel, Nancy, Nivis y Dulce, a mis suegros Rafael y Yamile, a mi cuñada Clara y a mi cuñado Luis. Agradecer a mi familia en la Habana. A mis primas Karina y Katia y a sus esposos, a mis tíos Armando y Maricela y toda la familia de plaza. Todos ellos siempre estuvieron ahí para cualquier necesidad, siempre estuvo su pregunta, "¿Cómo te va en la escuela, todo bien por allá?, estudia para que salgas bien" y cada vez que pasaba por donde ellos, me embutían la comida, y yo mencionaba que estaba a dieta. Agradecer a mis compañeros de aula, al antiguo grupo 4, a los que hoy están aquí y también a los que no se encuentran con nosotros a todas las chicas del grupo. Agradecer especialmente a Víctor, Eddy, Dariel, Elvis, Adrián, Luis, Ricardo, Gilberto, Asley, Osmar, Antonio, Karel, Maikel, el Pimi; a casi todo el paso de escalera de mi edificio todos ellos forman parte de la familia que se formó durante la carrera, de la cual estoy muy contento. Agradecer al tutor por la dedicación a nosotros durante el desarrollo de la tesis, a los profesores que influyeron en mi formación como ingeniero; a mi compañera de tesis. También agradecer a los asaos del Duty a los que les tiré MOAB, o simplemente dejé que ellos la tiraran. Especial agradecimiento al apartamento 15 107, Ariam (mi hijo), Enrique, Alejandro, el asao del Lachy, a Leobel. No faltaría menos mencionar al equipo aguerrido de la tienda panamericana de la UCI, compañeras y compañeros desde mis inicios en la universidad, a todos ellos muchas gracias. Y para todos aquellos que no he mencionado que han ayudado en mi formación como ingeniero muchas gracias.

Resumen

La seguridad e integridad de los datos de un sistema son el eslabón fundamental para el correcto funcionamiento del mismo. La creación y restauración de copias de seguridad favorecen en gran medida estos pilares para el desempeño de cualquier sistema. Actualmente en el sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1, la realización de estos procesos resulta engorroso para el usuario a la hora de salvar grandes volúmenes de datos o cuando se necesite migrar el sistema hacia otros servidores con mejores prestaciones, debido a que no se tiene un respaldo de la información contenida en el mismo. Esta deficiencia en el sistema dificulta la ejecución de dicha actividad de migración.

La presente investigación refleja el proceso de desarrollo de la herramienta informática para la creación y restauración de copias de seguridad, con el propósito de garantizar la seguridad e integridad de los datos en el sistema GDA Xabal eXcriba 3.1. Para lograr el objetivo propuesto se realizó una amplia revisión bibliográfica, que permitió elaborar la fundamentación teórica de las copias de seguridad. También se realizó un estudio del mecanismo utilizado en los sistemas de gestión documental y las herramientas informáticas gestoras de copias de respaldos, siendo seleccionada Duplicity como instrumento de apoyo para la implementación de la nueva herramienta informática, así como las herramientas y tecnologías a utilizar en la implementación.

El desarrollo de la aplicación fue guiado por la metodología de desarrollo de software XP, utilizando Java y Bash como lenguaje de programación. Dicha herramienta informática tiene un carácter iterativo, permitiendo la inserción de nuevas funcionalidades y la corrección de errores.

Palabras claves: Copias de seguridad, Sistemas de Gestión Documental, Herramientas informáticas de Gestión de Copias de seguridad, Sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1.

Índice	
Resumen	I
Introducción	6
Capítulo1: Fundamentación teórica de las copias de seguridad	10
1.1 Marco teórico de las copias de seguridad.	10
1.2 Sistemas homólogos de gestión documental.	11
1.3 Herramientas informáticas para la gestión de respaldos	12
1.4 Tendencias actuales de las copias de seguridad o respaldo	13
1.4.1 Metodología para realizar una copia de respaldo	14
1.5 Selección de la metodología de software y herramientas a emplear	16
1.5.1 Metodología de desarrollo de software	16
1.5.2 Lenguaje de programación	17
1.5.3 Entorno de Desarrollo Integrado	18
1.5.4 Herramientas CASE	19
1.6 Conclusiones parciales	20
Capítulo 2: Exploración y planeación de la aplicación	21
2.1 Modelo de dominio	21
2.2 Propuesta de solución	22
2.3 Exploración	23
2.3.1 Historias de Usuario	23
2.3.1 Requisitos No Funcionales	28
2.4 Planificación	29
2.4.1 Estimación de esfuerzos por Historias de Usuarios	29
2.4.2 Plan de Liberaciones	30
2.5 Plan de Iteraciones	31
2.5.1 Iteración 1	31
2.5.2 Iteración 2	31
2.5.3 Plan de duración de las iteraciones.	32
2.6 Conclusiones parciales	32
Capítulo 3: Diseño, codificación y pruebas de la aplicación	33
3.1 Diseño de la herramienta informática	33
3.1.1 Tarjetas CRC	33
3.1.2 Estructura de la herramienta informática	34
3.1.3 Arquitectura de la herramienta informática	35

3.1.5 Patrones de diseño	36
3.2 Codificación de la herramienta informática.....	36
3.2.2 Iteración 2	38
3.3 Pruebas sobre la herramienta informática.....	41
3.3.1 Pruebas unitarias	41
3.3.3 Pruebas de integración	42
3.3.4 Pruebas de aceptación.....	42
3.5 Resultados de las pruebas.....	44
3.6 Conclusiones parciales.....	45
Conclusiones generales	46
Recomendaciones.....	47
Referencias Bibliográficas.....	48
Bibliografías.....	51
Glosario de Términos.....	56
Anexos	57

Índice de figuras

Figura 1 Modelo de dominio	21
Figura 2 Esquema de la Propuesta de solución.	22
Figura 3 HU 1, 2, 3, 4.....	26
Figura 4 HU 5	27
Figura 5 HU 6	27
Figura 6 HU 7	28
Figura 7 Estructura de la herramienta informática.....	34
Figura 8 Arquitectura de la herramienta informática.....	35
Figura 9 Método para crear copias de seguridad	40
Figura 10 Método para la restauración de copias de seguridad	40
Figura 11 Resultado de la prueba unitaria al método Configurar en JUnit	42
Figura 12 Resultados de las pruebas	44
Figura 13 HU 5, 6	57
Figura 14 HU 6.....	57
Figura 15 HU 7.....	58
Figura 16 Método fecha	58
Figura 17 Método listar backup	59
Figura 18 Resultado de la prueba unitaria al método Get_restore_list en JUnit.	60
Figura 19 Resultado de la prueba unitaria al método GetProperties en JUnit.	60

Índice de tablas

Tabla 1 HU Solicitar nombre de la base de datos	23
Tabla 2: HU Solicitar usuario y contraseña de la base de datos	24
Tabla 3: HU Localizar dirección del sistema eXcriba	24
Tabla 4: HU 4 Elegir lugar para guardar la copia	24
Tabla 5: HU 5 Guardar configuración del sistema.....	25
Tabla 6: HU 6 Realizar copia de seguridad del sistema	25
Tabla 7: HU 7 Realizar restaura del sistema.....	25
Tabla 8: Ptos. de Estimación por Historias de Usuario	30
Tabla 9: Historias de Usuario por módulos.....	30
Tabla 10: Historias de Usuario por módulos	30
Tabla 11: Plan de entregas.....	31
Tabla 12: Plan de duración de las iteraciones.....	32
Tabla 13: Tarjeta CRC Settings.....	33
Tabla 14: Tarjeta CRC Execution	34
Tabla 15: HU 1 Solicitar el nombre de la base de datos	37
Tabla 16: HU 2 Solicitar usuario y contraseña de la base de datos	37
Tabla 17: HU 2 Solicitar usuario y contraseña de la base de datos	37
Tabla 18: HU3 Localizar dirección del sistema eXcriba	38
Tabla 19: HU 4 Elegir lugar para guardar la copia del sistema	38
Tabla 20: HU 5 Guardar configuración	38
Tabla 21: HU 6 Realizar copia de seguridad del sistema.....	39
Tabla 22: HU 7 Realizar restaura del sistema	39

Tabla 23: Caso de prueba Guardar configuración del sistema.....	43
Tabla 24: Caso de prueba Realizar copia de seguridad del sistema.....	43
Tabla 25: Caso de prueba Realizar restaura del sistema	44

Introducción

Con el avance de las Tecnologías de la Información y las Comunicaciones (TIC) comienzan a desarrollarse los sistemas de gestión documental. Estos sistemas tienen como objetivo mantener centralizada, organizada y estructurada la información [1]. Su implantación como elemento clave en la actividad humana, ha dejado de ser un simple método de archivo masivo para convertirse en una herramienta de análisis de la información y la gestión del conocimiento [2].

En la actualidad las organizaciones tienden a almacenar la totalidad de sus datos informáticos en sistemas. Estos datos son generados a través de la actividad interna que en la misma se desarrolla a lo largo de su ciclo de vida; siendo útiles y relevantes para la gestión diaria y la toma de decisiones. El proceso de transformación de la información supone un esfuerzo, lo que incrementa su valor, y como consecuencia, la pérdida de la misma llevará a la empresa a incurrir en elevados costes económicos.

Con el fin de evitar la pérdida de documentación, surgen las copias de seguridad que se refiere a las copias de datos, las cuales puedan utilizarse para restaurar el original después de una eventual pérdida de datos. Debido a esas contingencias, se hace necesario para cualquier institución la realización de copias de seguridad o respaldo para la recuperación de la documentación en caso de afectaciones.

Como parte de la estrategia de informatización de la sociedad cubana, debido al acelerado desarrollo de las TIC, el Centro de Informatización de la Gestión Documental (CIGED) radicado en la Universidad de las Ciencias Informáticas (UCI), es el encargado de desarrollar soluciones para la optimización del trabajo y mejoramiento de la calidad de la gestión documental. Una de estas soluciones es el sistema de Gestión de Documentos Administrativos Xabal eXcriba 3.1, el cual está soportado sobre la plataforma Alfresco en su versión comunitaria por las ventajas que brinda en el ámbito de la gestión documental y por su condición de soberanía tecnológica. Actualmente este producto informático se encuentra desplegado en varias entidades del país, lo que representa un logro para el centro de desarrollo, pero a la vez constituye una dificultad. La misma está sustentada en la carencia de un mecanismo económico, rápido y que permita mantener la seguridad e integridad de la información almacenada por el sistema de GDA Xabal eXcriba 3.1.

Esta carencia deriva problemas tanto para las organizaciones como para el centro de desarrollo CIGED. En el caso de las organizaciones no podrán efectuar migraciones tecnológicas de sus servidores o no podrán tener respalda su información sino solicitan un servicio de soporte al centro periódicamente; trayendo como consecuencia la pérdida de la información de los procesos realizados hasta el momento en el sistema y el CIGED estaría en la obligación de designar un grupo de personas que solamente se dedicaran a esta tarea, afectando significativamente su productividad.

Atendiendo a todo lo antes expuesto, se define como **problema a resolver** ¿Cómo garantizar la seguridad e integridad de los datos en el sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1? El **objeto de estudio** lo constituyen los procesos para garantizar el respaldo e integridad de los datos en los sistemas de Gestión Documental. En correspondencia con el objeto de estudio determinado, el **campo de acción** está enmarcado en el proceso de creación y restauración de copias de seguridad de datos en el Gestor de Documentos Administrativos Xabal eXcriba 3.1

Para el desarrollo de este trabajo se define como **objetivo general**: Desarrollar una herramienta informática para la creación y restauración de copias de seguridad en el Gestor de Documentos Administrativos Xabal eXcriba 3.1, que garantice la seguridad e integridad de los datos en el sistema.

Para ello se han trazado los siguientes **objetivos específicos**:

1. Analizar el marco teórico de la investigación para comprender como se desarrolla el proceso de creación y restauración de copias de seguridad en los sistemas de gestión documental.
2. Caracterizar la herramienta informática para la creación y restauración de copias de seguridad sobre el Gestor de Documentos Administrativos Xabal eXcriba 3.1.
3. Implementar la herramienta informática para la creación y restauración de copias de seguridad sobre el Gestor de Documentos Administrativos Xabal eXcriba 3.1.
4. Validar la herramienta informática para la creación y restauración de copias de seguridad sobre el Gestor de Documentos Administrativos Xabal eXcriba 3.1 a partir de la realización de pruebas.

Las tareas que se proponen para dar cumplimiento a los objetivos específicos expuestos anteriormente son las siguientes:

1. Análisis de los principales conceptos asociados a la creación de copias de seguridad en sistemas informáticos para obtener los fundamentos teóricos necesarios para el desarrollo de la solución.
2. Caracterización de las soluciones informáticas para la creación de copias de seguridad en sistemas informáticos.
3. Caracterización del proceso o metodología para llevar a cabo la creación y restauración de las copias de seguridad en sistemas informáticos.
4. Selección de las metodologías de software, herramientas informáticas y tecnologías para llevar a cabo el desarrollo de la herramienta informática para la creación y restauración de copias de seguridad para el GDA Xabal eXcriba 3.1.
5. Definición de las principales necesidades a resolver para el desarrollo de la solución.
6. Definición de la arquitectura de la herramienta informática para la creación y restauración de las copias de seguridad para GDA Xabal eXcriba 3.1.

7. Análisis de los patrones para el diseño e implementación de la herramienta informática para la creación de copias de seguridad para el GDA Xabal eXcriba 3.1.
8. Implementación de la herramienta informática para la creación y restauración de las copias de seguridad para el GDA Xabal eXcriba 3.1.
9. Caracterización de los métodos y técnicas que se emplean para realizar pruebas a un software.
10. Aplicación de pruebas de calidad a la herramienta informática para la creación y restauración de copias de seguridad para el GDA Xabal eXcriba 3.1.

Métodos de investigación

La metodología de la presente investigación asume como principal criterio metodológico la concepción dialéctico materialista, garantizando así el desarrollo y evolución de este trabajo. Para el desarrollo de la investigación se utilizaron los métodos empíricos y teóricos.

Métodos teóricos

Analítico - Sintético: Para el desarrollo del trabajo de diploma se utilizó este método porque permite el estudio y revisión de la bibliografía necesaria para tener una mejor comprensión del funcionamiento de las herramientas de gestión de copias de seguridad, su caracterización para la posterior selección de una herramienta informática gestora de copias de respaldos que sirva de apoyo para la implementación de la nueva herramienta informática a desarrollar.

Modelación: Se utilizó en el diseño del modelo de dominio que representa el funcionamiento del proceso del negocio en el sistema GDA Xabal eXcriba 3.1 y la esquematización del funcionamiento del proceso de creación y restauración de copias de respaldos de la nueva herramienta informática como propuesta de solución del problema planteado.

Métodos empíricos

Observación: Permite obtener las características del proceso de creación y restauración de copias de respaldos. La observación, como procedimiento se ha utilizado en distintos momentos de la investigación: en su etapa inicial se utiliza en el diagnóstico del problema a resolver y es de gran utilidad para la captura de los requisitos de la nueva herramienta informática, además de la información necesaria para su desarrollo en la fase de diseño e implementación.

Estructuración del trabajo de diploma

El presente trabajo de diploma se ha estructurado en introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas, bibliografía y glosario de términos, para una mejor comprensión se ofrece a continuación una breve descripción de cada capítulo.

Capítulo 1: Fundamentación teórica de las copias de seguridad. En este capítulo se realiza un estudio del estado del arte de los sistemas de gestión documental y las herramientas informáticas de gestión de copias de seguridad. Se selecciona la herramienta informática a emplear como instrumento de apoyo para la implementación del producto. Además, se seleccionan las metodologías, herramientas y tecnologías a utilizar en el desarrollo de la nueva herramienta informática.

Capítulo 2: Exploración y Planificación de la aplicación. En este capítulo se hace referencia a los conceptos relacionados con el negocio, además se utiliza como apoyo auxiliar un modelo de dominio que los refleja de manera explícita. Se presentan las Historias de Usuario descritas en la fase de Exploración y la Planificación realizada a partir de éstas, como parte del trabajo llevado a cabo en la fase de igual nombre.

Capítulo 3: Diseño, Codificación y Pruebas de la aplicación. En este capítulo se describen los aspectos asociados al diseño, la implementación y las pruebas de la aplicación. Se presentan artefactos como las tarjetas CRC (Clase, Responsabilidad, Colaboración), la arquitectura de la herramienta informática y las tareas de programación, con el fin de facilitar la implementación. También se detallan las pruebas funcionales realizadas para verificar la correspondencia según lo acordado con el usuario.

Capítulo 1: Fundamentación teórica de las copias de seguridad

En el presente capítulo se exhiben los fundamentos teóricos de los distintos tipos de copias de seguridad que existen, así como los conceptos relacionados con las mismas; sirviendo como base teórica para la solución del problema en cuestión. Además, se aborda la investigación realizada sobre los sistemas de gestión documental para conocer cómo desarrollan el proceso de realización de las copias de respaldos. Se realiza un estudio acerca de las herramientas informáticas existentes para la creación y restauración de copias de seguridad. También se efectúa el análisis de la metodología a seguir para efectuar el proceso de creación y restauración de un respaldo, así como la comprobación de la integridad de los datos guardados.

1.1 Marco teórico de las copias de seguridad.

Muchas son las razones por las que día a día se pierde información, las más comunes son: mal funcionamiento de hardware o software, errores humanos, virus o desastres naturales[3]. Con el fin de mitigar dichas pérdidas de información existen las copias de seguridad o las también llamadas copias de respaldos. Su objetivo principal es evitar la pérdida de datos y poder recuperar la información después de haber ocurrido algún desastre, a partir de copias realizadas con anterioridad. El término copia de seguridad está asociado con el término restauración, ya que el mismo es la acción de devolver los datos perdidos a su ubicación original[4].

Copias de seguridad o respaldos

Existen un sin número de definiciones acerca de las copias de respaldo, aunque todas convergen en la misma idea. A continuación, se muestran varias de estas definiciones obtenidas de diferentes fuentes:

- En informática, el backup o copia de seguridad, es la copia total o parcial de información importante como respaldo frente a eventualidades. Debe ser guardada en un soporte almacenamiento diferente del original, para evitar que un fallo en el mismo pueda estropear el original y la copia. Por ejemplo, si se tienen archivos importantes en un disco duro de la computadora, lo ideal sería hacerles una copia de seguridad almacenada en un pendrive o en una unidad de almacenamiento en la nube. Los respaldos se utilizan para contar con una o más copias de información considerada importante y así poder recuperarla en el caso de pérdida de la copia original [5].

- Copia de datos de forma que estén disponibles en caso de que un fallo produzca la pérdida de los originales. Esta sencilla acción evita numerosos y a veces irremediables problemas si se realiza de forma habitual y periódica [6].

- Hacer una copia de seguridad o copia de respaldo (Backups en inglés) se refiere a la copia de datos de tal forma que estas copias adicionales pueden restaurar un sistema después de una pérdida de información [7].
- Copia de seguridad es la actividad de copiar archivos a bases de datos a fin de que puedan ser conservadas en casos de fallo del equipo o de otra catástrofe. La copia de seguridad suele ser una parte rutinaria de la operación de grandes empresas con supercomputadoras, así como los administradores de las empresas más pequeñas con menos recursos. Para los usuarios de computadoras personales, la copia de seguridad también es necesaria pero no se realiza muy a menudo. La recuperación de los archivos de la copia de seguridad se denomina: restablecimiento [8].

De las definiciones anteriormente expuestas, la que más se ajusta al problema a resolver es la definición que se refiere a hacer copias de seguridad mediante backups, ya sean locales o no. La idea fundamental es utilizar un sistema de respaldo que realice copias de seguridad con una configuración sencilla y segura.

1.2 Sistemas homólogos de gestión documental.

Hoy en día, existen en el mundo varios sistemas de gestión documental que son empleados por las empresas para manejar y mantener un control sobre la información en criterio de documentos. En el presente trabajo se realizó un estudio de estos sistemas enfocado en el mecanismo de creación y restauración de las copias de respaldos, con el objetivo de conocer cómo estos sistemas de gestión documental realizan el proceso de creación y restauración de las copias de respaldos. De los ejemplares encontrados se tomó una muestra de siete sistemas para realizar el análisis, los cuales se citan a continuación: eFileCabinet[9], Alfresco One[10], docStar[11][12], Ademero[13][14][15], SmartFile[16][17], Athento[18], y Knowledge Tree[19]. Por tener un grupo de características similares a las del software GDA Xabal eXcriba 3.1. Éstos cuentan con un mecanismo para realizar la copia y restauración de datos; sin embargo, no se puede hacer uso de dicha funcionalidad como apoyo para la implementación de la nueva herramienta informática porque varios de los sistemas tienen licencia privativa o no se ha obtenido la información suficientemente detallada por parte de los proveedores, para conocer cómo estos realizan el proceso de creación y restauración de copias de respaldos.

Por tal motivo, se decidió buscar como segunda alternativa realizar un estudio de las diferentes herramientas informáticas gestoras de copias de seguridad que existen, con el objetivo de profundizar en los conocimientos acerca de la ejecución de los procesos de creación y restauración de respaldos. Además, encontrar un gestor de copias que sirva de apoyo para la implementación de la nueva herramienta informática.

1.3 Herramientas informáticas para la gestión de respaldos

Con el objetivo de comprender cómo las herramientas informáticas gestoras de copias de seguridad realizan el mecanismo de creación y restauración de copias de respaldos, se realizó una investigación acerca de las mismas y se detectaron varias herramientas informáticas capaces de optimizar estos procesos. De todos los ejemplares encontrados se realizó un análisis sobre Rsync, Duplicity, Dump/restore y LuckyBackup basado en sus principales características, tales como: plataformas soportadas, interfaces disponibles, tipos de respaldos que realizan y cifrado de datos.

Rsync: es una herramienta que permite replicar un sistema de archivos manteniendo los propietarios y permisos originales. Esta herramienta es capaz de adaptarse para hacer copias de seguridad de tipo completa e incremental, además de permitir su restauración [20]. Dicho gestor ofrece varias prestaciones de soporte para copiar enlaces, ficheros de dispositivo, propietarios, grupos y permisos.

Duplicity: es una herramienta que permite realizar copias de seguridad incremental de archivos y directorios, cifrarlos a través de GnuPG y subirlas a un servidor de archivos remoto o local, utilizando para la transmisión de datos ssh/scp/ftp/rsync. Soporta todos los permisos de sistemas[21].

LuckyBackup: es una herramienta para realizar backups y sincronizar archivos basada en rsync. Posibilita la sincronización de carpetas, manteniendo los archivos más recientes de ambas. Permite excluir datos de una tarea para solo incluir los datos específicos. Soporta conexiones remotas. Ofrece la posibilidad de ejecutar comandos antes y después de una tarea. Tiene capacidad de restauración a cualquier ruta de acceso [22]. Sus backups son por defectos incrementales[23]. Permite añadir opciones de seguridad y cifrado de datos. Brinda la posibilidad de crear uno o varios perfiles para la gestión de backups, permitiendo añadir carpetas de origen y destino, las cuales pueden ser locales o remotas [24].

Dump/restore: es la herramienta clásica para realizar backups en entornos Unix. Permite realizar backups sobre máquinas remotas directamente desde líneas de comandos. Examina los archivos en un sistema de archivos, determina cuáles deben ser respaldados y copia esos archivos en un disco especificado, ya sea una cinta u otro medio de almacenamiento. Permite recuperar archivos individuales y subárboles de directorio de copias de seguridad completa o parcial en el modo de interactivas[25].

Luego de analizar dichas herramientas informáticas, se decidió emplear Duplicity como apoyo para la implementación del nuevo producto; después de conocer que la misma es una herramienta multiplataforma que permite realizar copias de seguridad incremental y total. Brinda la posibilidad

de firmar las copias como muestra de autenticidad del autor, característica de la cual carecen las herramientas informáticas antes mencionadas. Además, ofrece mejoras en cuanto al uso de espacio y ancho de banda, pues utiliza internamente el algoritmo rsync. Ofrece formato de archivo estándar y posibilita la selección de protocolo remoto y almacenamiento local. Permite hacer restauraciones de archivos de una fecha específica, brinda soporte de dispositivos y trabajo en red.

El resto de las herramientas informáticas analizadas fueron descartadas porque presentan deficiencias; siendo estas deficiencias algunos de los aspectos fundamentales para la implementación de la aplicación, con el fin de garantizar la seguridad de los datos. Dichas desventajas se explican a continuación:

Rsync presenta deficiencia en cuanto al cifrado de los datos, dado a que solo ofrece seguridad en los datos para la autenticación; donde utiliza un protocolo basado en MD4 de 128 bits. Sin embargo, no cifra la información transmitida, la cual puede ser corrompida durante la realización del proceso de creación de la copia. Una de las desventajas que tiene Dump/ restore con respecto a Duplicity es solo realiza respaldos para los sistemas de archivos ext2 y ext3. La principal deficiencia que presenta la aplicación LuckyBackup es que solo realiza copias incrementales, característica que no favorece al nuevo producto a desarrollar

1.4 Tendencias actuales de las copias de seguridad o respaldo

Actualmente existen muchos y diferentes tipos de dispositivos para almacenar datos que son útiles para hacer backups. El medio elegido para la copia depende de múltiples factores como: tipo y valor de la información que se desea guardar, costos del soporte, del copiado, del almacenamiento y su posterior recuperación, métodos de acceso a la información guardada, tiempo de vida útil del soporte, ventajas y desventajas del mismo[5]. También existen diferentes formas en que estos dispositivos puede ser dispuestos para proveer repetitividad, seguridad en los datos y facilidad de traslado, antes de que los datos sean enviados a su lugar de almacenamiento se deben seleccionar y extraer.

Aparejado a los diferentes tipos de dispositivos utilizados para crear copias de respaldos también se han desarrollado disímiles técnicas con el fin de optimizar el procedimiento de efectuar las copias. Estos procedimientos incluyen otros mecanismos que encierran nuevas optimizaciones para trabajar con archivos abiertos y fuentes de datos en uso. Además, contienen procesos de compresión, cifrado y procesos de duplicación, entendiéndose este último una forma específica de compresión donde los datos son eliminados.

Las copias de respaldos son útiles fundamentalmente para dos tareas: primero para recuperarse de una catástrofe informática y segundo para recuperar un porcentaje de la cantidad de archivos que han sido corrompidos o eliminados de forma accidental. Debido a que los sistemas de copia de seguridad contienen como mínimo una copia de todos los datos que vale la pena salvar, se debe

tener en cuenta los requerimientos de almacenamiento, la organización del espacio de almacenamiento y la administración del proceso de efectuar la copia. Éste último es utilizado para resolver disímiles problemas entre los que se encuentran restaurar un ordenador a un estado operacional después de un desastre (copias de seguridad del sistema) y restaurar un pequeño número de ficheros después de que hayan sido borrados o dañados accidentalmente (copias de seguridad de datos).

Una copia de respaldo puede realizarse sobre los datos, en los cuales se incluyen también archivos que formen parte del sistema operativo, así las copias de seguridad suelen ser utilizadas como la última línea de defensa contra pérdida de datos y se convierten por lo tanto en el último recurso a utilizar; teniendo como objetivo en un sistema informático preservar cierta capacidad de recuperación de la información ante posibles pérdidas. Dicha capacidad puede convertirse en algo muy relevante, e incluso hasta crítico para muchas organizaciones. Existen empresas que han desaparecido del mercado ante la imposibilidad de recuperar los datos de sus sistemas tras la ocurrencia de una contingencia de seguridad delicada.

1.4.1 Metodología para realizar una copia de respaldo

Para trazar cualquier estrategia de creación de copias de seguridad se hace imprescindible comenzar con el concepto de almacén de datos. Los datos de la o las copias deben ser guardados de alguna manera y seguramente son organizados con algún criterio; pudiéndose hacer uso desde una hoja de papel con una lista de las cintas de la copia de seguridad y las fechas en que fueron hechas, hasta un sofisticado programa con una base de datos relacional. Cada uno de los distintos almacenes de datos tiene sus ventajas, y está muy relacionado con el esquema de rotación de copia de seguridad escogido. Para ello, se han definido las diversas políticas para realización de copias de seguridad o respaldo; pero el presente trabajo hará uso de la metodología planteada por la IBM para la realización de copias de seguridad debido a que sus métodos están definidos de manera general para crear cualquier tipo de copia, ya sean respaldos para los archivos de un sistema, directorios, información en criterio de documentos u otros. Al mismo tiempo, dichas políticas se ajustan al problema que se pretende resolver.

1. Se debe definir el ámbito de aplicación: se clasifican los datos que se van a incluir en la copia. A su vez se establece el nivel de protección que hay que asignarles a las copias realizadas (es decir, empleo de criptografía, confinamiento de los backups o cifrados). Dicha especificación del tipo de datos puede influir en los siguientes apartados (en función, por ejemplo, de si se trata de datos de usuarios, copia del sistema operativo, copia de aplicaciones). La manipulación de los datos guardados en la copia de seguridad es una práctica habitual para optimizar tanto los procesos de copia como el almacenamiento.

- **Compresión:** La compresión es un método eficaz para disminuir el espacio de almacenaje necesario y de ese modo reducir el coste.

- **Redundancia:** Cuando varios sistemas guardan sus copias de seguridad en el mismo sistema de almacenamiento, existe la posibilidad de redundancia en los datos copiados. Si tenemos estaciones con el mismo sistema operativo compartiendo el mismo almacén de datos, existe la posibilidad de que la mayoría de los archivos del sistema sean comunes. El almacén de datos realmente sólo necesita almacenar una copia de esos ficheros para luego ser utilizada por cualquiera de las estaciones, esta técnica puede ser aplicada al nivel de ficheros o incluso al nivel de bloques de datos, reduciendo el espacio utilizado para almacenar.
 - **Des-duplicación:** Algunas veces las copias de seguridad están duplicadas en un segundo soporte de almacenamiento. Esto puede hacerse para cambiar de lugar imágenes, para optimizar velocidades de restauración o incluso para disponer de una segunda copia a salvo en un lugar diferente o en soportes de almacenamiento diferentes.
 - **Cifrado:** La alta capacidad de los soportes de almacenamiento desmontables implica un riesgo de perderse o ser robados, si se cifra la información de estos soportes se puede mitigar el problema, aunque esto presenta nuevos inconvenientes: Primero cifrar es un proceso que consume mucho tiempo de CPU y puede bajar la velocidad de copiado, en segundo lugar, una vez cifrados los datos, la compresión es menos eficaz.
2. En el segundo paso se definirá la periodicidad de las copias que se van a realizar: No existe un criterio definitivo que indique la periodicidad con la que deben realizarse las copias de seguridad, aunque sí algunas consideraciones orientativas en este sentido. De manera general, si los documentos con los que se trabaja se modifican varias veces cada día, es recomendable realizar la copia como mínimo también a diario, y en algunos casos, creando copias de respaldo provisionales en otras unidades de almacenamiento.
 3. Por tal motivo se define como tercer paso la elección del soporte para la copia. Existen varios tipos de dispositivos de almacenamiento que también pueden ser utilizados para la realización de un backup, tales como: cintas, CD, DVD, BluRay, discos virtuales (proporcionados por Internet) y discos duros.
 4. Seleccionar el método de copia de seguridad a emplear. Existen varias clasificaciones de los tipos de copias, las cuales se describen a continuación y se pueden hacer combinaciones entre ellas:
 - **Desestructurado:** Un almacén desestructurado podría ser simplemente una pila de disquetes o CD-R con una mínima información sobre qué ha sido copiado y cuándo. Ésta es la forma más fácil de implementar, pero ofrece pocas garantías de recuperación de datos.
 - **Copia Completa:** Copia de respaldo donde se protegen todos los archivos definidos en el plan de copias.
 - **Copia Incremental:** es aquella donde solo almacenará la información que haya sido modificada desde la última copia de seguridad realizada, ya sea completa, diferencial o incremental.

- **Completa + Incremental:** En primer lugar, se realiza la copia de seguridad completa del sistema, más tarde se realiza una copia de seguridad incremental, es decir, sólo con los ficheros que se hayan modificado desde la última copia de seguridad.
 - **Copia diferencial:** es un proceso donde se copian solamente los datos que han sido modificados desde la última copia de seguridad completa que se realizó.
 - **Espejo + Diferencial:** Un almacén de tipo espejo + diferencial inversa es similar al almacén completo-incremental la diferencia está, en que ofrece un espejo que refleja el estado del sistema a partir de la última copia y un historial de copias diferenciales.
 - **Protección continua de datos:** En este modelo no se realizan copias periódicas. El sistema inmediatamente registra cada cambio en el sistema anfitrión. Reduce al mínimo la cantidad de información perdida en caso de desastre.
 - **Sintética (synthetic backup):** Esta tecnología permite crear una nueva imagen de copia de respaldo a partir de copias de respaldo anteriormente completas y posteriores incrementales. Es de gran utilidad sobre todo en redes de almacenamiento (SAN) ya que no es necesaria la participación del host/nodo final, quitándole mucha carga de proceso.
5. Determinar la ubicación final de la(s) copia(s).
 6. Aplicar protección de los datos por duplicación de discos. Esto ayuda a mantener los datos disponibles ya que se tiene dos copias de los datos en dos unidades de discos aparte.
 7. Elegir los responsables y asignar responsabilidades (tanto de realizar las copias como de llevar a cabo la comprobación de la integridad de los datos almacenados).

De los aspectos citados para la realización de una copia de respaldo, se utilizaron los pasos 1, 3, 4, 5 y 7. Estos pasos sirven de apoyo para la implementación de la nueva herramienta informática; permitiendo establecer un orden lógico en las funcionalidades de la aplicación. Además, dichos aspectos, brindaron información para el análisis lógico de la codificación de algunas de las funcionalidades que contiene el nuevo producto.

1.5 Selección de la metodología de software y herramientas a emplear

Con la finalidad de darle solución al objetivo general del presente trabajo de diploma, se hace necesario prescindir de algunos aspectos. Para ello se definirá la metodología de software, las herramientas informáticas y tecnologías para llevar a cabo el desarrollo de la herramienta informática para la creación de copias de seguridad para el sistema GDA Xabal eXcriba 3.1.

1.5.1 Metodología de desarrollo de software

Las metodologías de desarrollo de software son el conjunto de procedimientos, técnicas, herramientas y un soporte documental, que ayuda a los desarrolladores a realizar nuevo software. Estas definen quién debe hacer qué, cuándo y cómo debe hacerlo para obtener los distintos

productos parciales y finales. Se clasifican en dos tipos: tradicionales y ágiles. Aunque el proceso de desarrollo de software incluye otros aspectos importantes y determinantes, es precisamente la metodología utilizada, el elemento rector a lo largo del mismo. Por tal motivo es que se debe analizar, de acuerdo a las características del software a desarrollar y del equipo de desarrollo, cuál es la más se ajusta al proyecto [26]. En tal sentido se propone emplear la metodología XP para el desarrollo del trabajo de diploma.

Tiene como base la simplicidad y como objetivo principal la satisfacción del cliente [27]; para lograrlo se deben tomar en cuenta cinco valores fundamentales: comunicación, simplicidad, retroalimentación, valentía y respeto [28]. Consta de 6 fases para el ciclo de desarrollo de un software en muchas de las fuentes consultadas: Exploración, Planificación de la Entrega (Release), Producción, Mantenimiento y Muerte del Proyecto [27]. Pero el presente trabajo estará guiado por la bibliografía Ingeniería del software. Un enfoque práctico de Roger Pressman, en la cual se plantea en su séptima edición que XP tiene 4 etapas para el desarrollo de software: Planeación, Diseño, Codificación y Pruebas; donde la fase de Exploración se encuentra intrínseca en la etapa de Planeación.

Se caracteriza por ser una metodología ágil para pequeños y medianos equipos, desarrollando software cuando los requerimientos son ambiguos o rápidamente cambiantes; asume el cambio como algo natural. En XP se realiza el software que el cliente solicita y necesita, en el momento que lo precisa, alentando a los programadores a responder a los requerimientos cambiantes que plantea el cliente en cualquier momento. Esto es posible porque está diseñado para adaptarse en forma inmediata a los cambios, con bajos costos asociados, en cualquier etapa del ciclo de vida [29].

Se propone su uso, debido a que el equipo de trabajo es pequeño, el período de tiempo para la implementación de la aplicación es de menos de 6 meses por lo que se considera un período corto; donde XP facilita en gran medida el trabajo porque no existe la necesidad de generar un gran número de artefactos para el desarrollo de la nueva herramienta informática. No existe dependencia de documentación anterior para la implementación de la misma.

Además, el producto a desarrollar está basado en el uso de nuevas tecnologías por lo que se suman los riesgos técnicos asociados a ellas, la planificación se realiza a corto plazo porque está sujeta a las necesidades de la producción donde se exigen entregas en breves intervalos, lo cual es perfectamente posible gracias a la naturaleza iterativa e incremental que posee dicha metodología.

1.5.2 Lenguaje de programación

Un lenguaje de programación es un lenguaje artificial que puede ser usado para controlar el comportamiento de una máquina, especialmente una computadora. Estos se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán interpretadas[30]. Para llevar a cabo la implementación se decide escoger como lenguaje de programación:

Java 7: Java es un lenguaje de alto nivel desarrollado por Sun Microsystems. La sintaxis de Java es similar a C++, pero es estrictamente un lenguaje de programación orientado a objetos. Los programas de Java son multiplataforma [31]. Java presenta características que lo convierten en un lenguaje seguro, estándar y de alto nivel, algunas de las principales características se muestran a continuación: orientado a objetos, pues soporta las tres características propias de la orientación a objetos: encapsulación, herencia y polimorfismo. También es un lenguaje distribuido porque proporciona las librerías y herramientas para que los programas puedan distribuirse, es decir, que se puedan ejecutar en varias máquinas interactuando [32]. El equipo de desarrollo decidió implementar la nueva herramienta informática sobre este lenguaje de programación por su versatilidad y la posibilidad que brindan sus aplicaciones de poder ser empleadas independientemente al sistema operativo donde se pretendan utilizar. De manera general el equipo de desarrollo dispone de poco tiempo para llevar a cabo la implementación de la nueva herramienta informática y no se puede dedicar horas extras para estudiar otros lenguajes de programación. Por ende, se decide desarrollar el producto en Java, dado a que se tiene conocimiento del lenguaje porque ha sido utilizado en el transcurso de la especialidad para desarrollar varias aplicaciones, lo que facilita su uso.

Bash 4.3: es un lenguaje que acepta y ejecuta comandos. También soporta construcciones de programación que permiten la construcción de comandos complejos (scripts) a partir de partes más pequeñas. El bash es uno de los numerosos shells disponibles para Linux. Incorpora características del *shell Korn (ksh)* y del *shell C (csh)*, y ha sido pensado para ser compatible con el *shell POSIX* [33]. Este lenguaje se utilizará como lenguaje intermedio para establecer una comunicación entre Java y el sistema operativo a través líneas de comandos. Permitirá la implementación del código fuente de la herramienta informática.

1.5.3 Entorno de Desarrollo Integrado

Un entorno de desarrollo integrado (IDE) es un entorno de programación que ha sido empaquetado como un programa de aplicación, por lo general consiste en un editor de código, un compilador, un depurador, y una interfaz gráfica de usuario (GUI) constructor. El IDE puede ser una aplicación independiente o puede ser incluido como parte de una o más aplicaciones existentes y compatibles [34]. Con el fin de llevar a cabo el desarrollo de la herramienta informática para la creación y restauración de copias de seguridad se decide escoger como IDE de programación:

NetBeans IDE 8.0.1: es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Es una herramienta que está soportada en Java. El mismo es un producto libre y gratuito sin restricciones de uso [35]. Permitirá la creación de interfaces gráficas para la herramienta informática y facilitará el manejo de toda la lógica del negocio. Permitirá la

conexión con la herramienta Duplicity para ejecutar las operaciones de creación y restauración de copias de respaldos. Durante el transcurso de la carrera el equipo de desarrollo ha implementado varias soluciones haciendo uso del mismo, por lo que se tiene dominio de dicho entorno de desarrollo y esto facilita en gran medida la codificación de la herramienta informática a desarrollar.

Sublime Text 3.1: es un editor de código al estilo TextMate, Kate o Redcar, su interfaz es limpia e intuitiva y soporta el uso de Snippets; Plugins y sistemas de construcción de código. Consta de varias características entre las que resaltan minimapa, multiselección, acceso rápido a líneas o archivos, paletas de comandos[36], autocompletado y marcado de llaves. Debido a las características citadas se define utilizar este editor para agilizar la implementación de scripts, además SublimeText facilita el trabajo al programador.

1.5.4 Herramientas CASE

Las herramientas CASE son un conjunto de herramientas y métodos asociados que proporcionan asistencia automatizada en el proceso de desarrollo del software a lo largo de su ciclo de vida. Fueron desarrolladas para automatizar esos procesos y facilitar las tareas de coordinación de los eventos que necesitan ser mejorados en el ciclo de desarrollo de software[37]. Como herramienta CASE se decide utilizar Visual Paradigm for UML 8.0 para el desarrollo de la herramienta informática para la creación y restauración de copias de respaldo.

Visual Paradigm for UML 8.0: es una herramienta de diseño de software diseñado para proyectos de software ágil. Soporta varios estándares de modelado entre los cuales se destaca UML. Facilita la creación de software y sistemas que se destacan en la experiencia del usuario mediante el apoyo a la utilización eficaz de identificación de casos, recopilación de requisitos, el flujo de los acontecimientos, generación de especificación de requisitos, entre otros[38]. Además de las ventajas expuestas, es válido resaltar que se selecciona Visual Paradigm for UML 8.0 porque permitirá la creación de un modelo de dominio de la herramienta informática a implementar, dado a que se hará uso del modelado UML para diseñar dicho modelo de dominio. Durante el estudio de la especialidad, en las disciplinas de ingeniería de software y gestión de software se ha hecho uso de Visual Paradigm for UML para elaborar disímiles proyectos, lo que hace que ésta sea una herramienta muy conocida por el equipo de desarrollo y se tenga dominio de la misma. Por tal motivo, se hará uso de dicha herramienta case; independientemente de tener licencia privativa y no estar distribuida para el sistema operativo Linux en el que está desplegado el software GDA Xabal eXcriba.

1.6 Conclusiones parciales

En el presente capítulo se realizó un análisis acerca del proceso de creación y restauración de copias de respaldos sobre los sistemas de gestión documental, donde el mismo estaba basado en conocer cómo los sistemas de gestión documental realizan el proceso de creación y restauración de copias y encontrar elementos que sirvieran de apoyo para el desarrollo del nuevo producto. En tal sentido, posteriormente se realizó un estudio de las diferentes herramientas informáticas encargadas de la gestión de copias de seguridad, obteniéndose como resultado:

- La comprensión de los procesos de creación y restauración de copias de respaldo de dichas herramientas informáticas.
- La selección de una herramienta informática como Duplycity que facilita la implementación de la nueva herramienta informática para la creación y restauración de copias de respaldos. Teniendo en cuenta sus características y funcionalidades, y que pueden ser usadas para el desarrollo de la nueva aplicación.
- Duplycity es lo más factible para el desarrollo de la nueva herramienta informática para la creación y restauración de copias de respaldos, dado a que ofrece varias ventajas que favorecen la seguridad de la información y facilidad de uso.

También se realizó un estudio de las metodologías de desarrollo, los lenguajes y las herramientas necesarias para la implementación del sistema de copias de seguridad. El estudio realizado posibilitó entender que con el uso de las tecnologías y herramientas seleccionadas para el desarrollo del producto final, facilitó el desarrollo de una herramienta informática para la creación y restauración de copias de seguridad soportado por el lenguaje de programación Java con código Bash como lenguaje de programación intermedio; haciendo uso de NetBeans y Sublime Text como plataforma de desarrollo con la utilización de Visual Paradigm for UML 8 para el diseño del modelo de dominio; los que facilitaron la codificación del software, guiado por XP como metodología de desarrollo de software.

Una vez definida la plataforma de la nueva herramienta informática a implementar, se procede a detallar las características de la aplicación.

Capítulo 2: Exploración y planeación de la aplicación

El presente capítulo tiene como objetivo describir la propuesta de solución para el problema a resolver antes planteado y precisar las características que debe poseer la herramienta informática para la creación de copias de seguridad a implementar.

A partir de la metodología seleccionada se hace referencia a los puntos relacionados con las fases de Exploración y Planificación correspondiente a la primera etapa Planificación del proyecto, en la que se detallan las Historias de Usuario para establecer posteriormente el orden en que serán implementadas atendiendo a su prioridad, con el fin de organizar el ciclo de vida del producto. Para ello, se requiere de la definición de una representación del modelo de dominio de la aplicación como artefacto complementario.

2.1 Modelo de dominio

Un modelo del dominio es una representación de las clases conceptuales o entidades del mundo real. Esto ayuda a los usuarios, clientes y desarrolladores e interesados; a utilizar un lenguaje común para poder entender el contexto en que se desarrolla el sistema. Es decir, captura los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema y no incluyen las responsabilidades de las personas que ejecutan las actividades[39].

Con el objetivo de complementar el entendimiento de la dinámica asociada se hace uso de un modelo de dominio, aunque la metodología XP no propone artefactos, pero en este sentido es flexible y puede contrastarse con algunos que simplifiquen y apoyen el proceso. Basado en esta definición, se muestra a continuación el modelo de dominio del componente para realizar el proceso de copias de seguridad para el sistema de Gestión de Documentos Administrativos Xabal eXcriba v3.1.

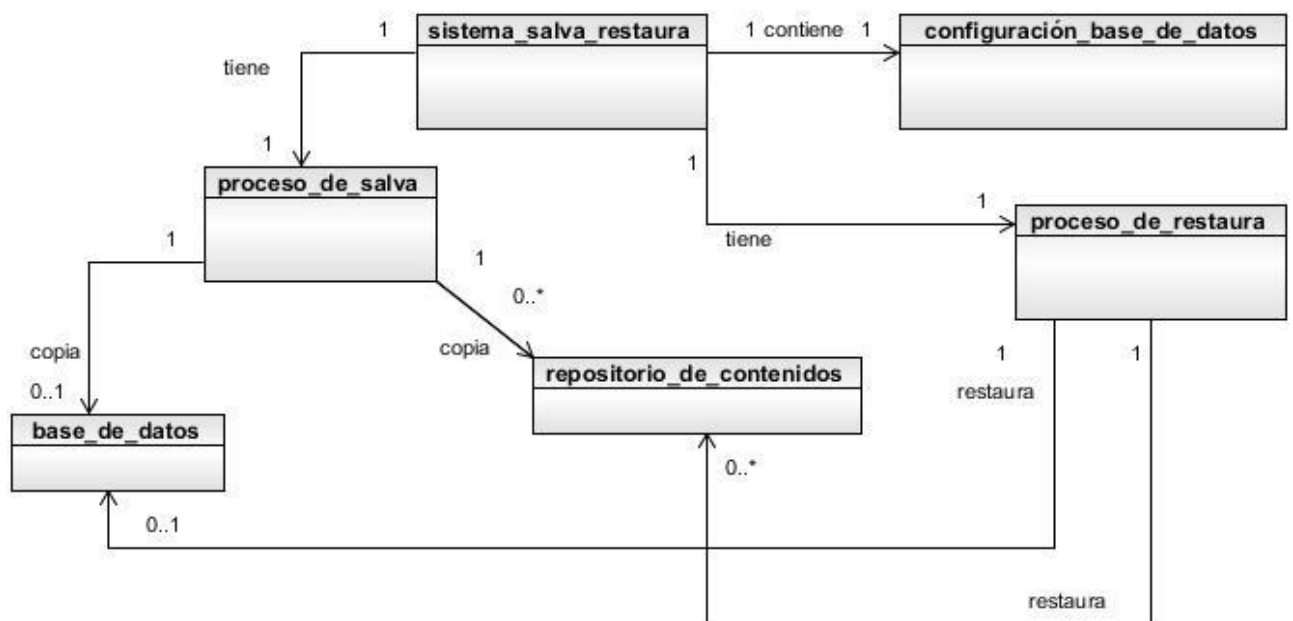


Figura 1 Modelo de dominio

El concepto principal de este modelo lo constituye el sistema_salva_restaura, el cual representa el macro-proceso de la creación y restauración de copias de respaldos y contiene tres entidades asociadas de manera única y está directamente conectada a ellas. Uno de los conceptos importante asociado al sistema_salva_restaura es el concepto proceso_de_salva, con quien tiene una asociación única y refleja el procedimiento del mecanismo de realización de las copias. El mismo está vinculado a otros tres conceptos formados por al menos una copia de la base de datos y cero copias del repositorio de contenidos. Otro concepto importante asociado al sistema_salva_restaura lo constituye el proceso_de_restaura que ilustra la ejecución del proceso de restauración del sistema y está conectado de manera única con la base de datos y los repositorios de contenidos. Por último, se encuentra el concepto configuración que aborda todos los datos que se manipularán para realizar la configuración de los parámetros de la herramienta a implementar y tiene una conexión única con el concepto principal sistema_salva_restaura.

2.2 Propuesta de solución

El presente trabajo de diploma pretende desarrollar una herramienta informática que permita realizar la creación y restauración de las copias de seguridad del sistema GDA Xabal eXcriba 3.1.

La herramienta informática que se pretende implementar para la creación y restauración de copias de seguridad, contará con una primera instalación del sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1; soportado por el Gestor de Contenidos Empresariales ECM Alfresco en su versión 4.2f. Esta nueva herramienta informática será la encargada de crear las copias de la Base de Datos y el Repositorio de Contenidos; almacenando los mismos en un punto común denominado backup. Posteriormente las copias realizadas a la Base de Datos y el Repositorio de Contenidos guardadas en el punto común, podrán ser restauradas a una nueva instalación del sistema GDA Xabal eXcriba 3.1.

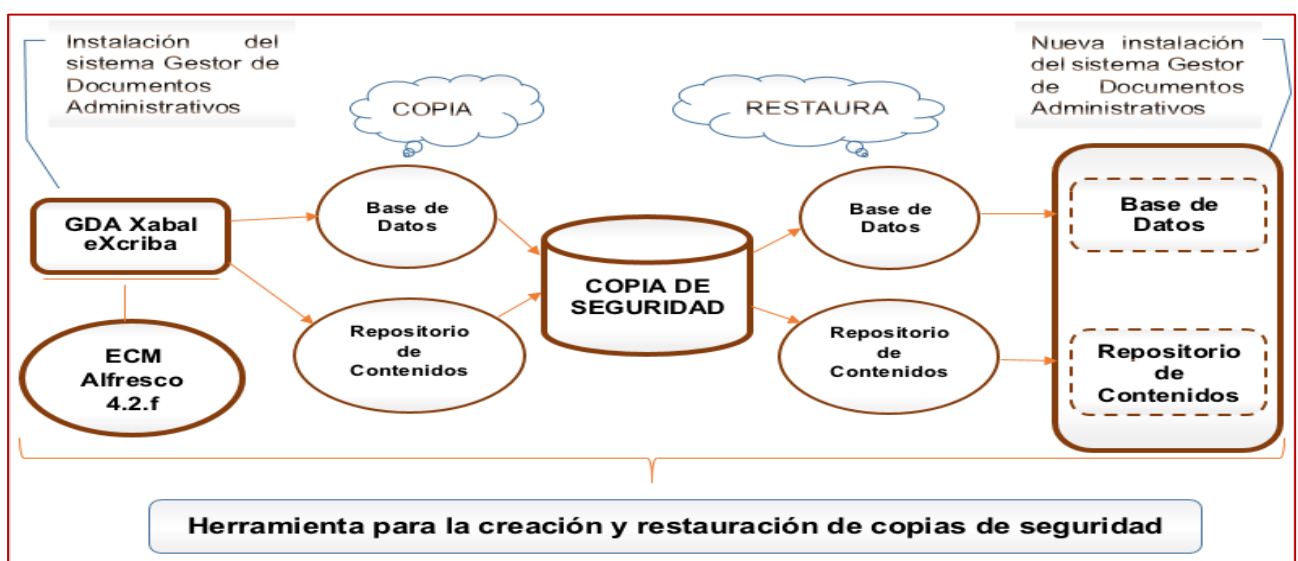


Figura 2 Esquema de la Propuesta de solución.

2.3 Exploración

El ciclo de vida ideal de un proyecto realizado con XP se inicia con la fase de Exploración, al final de la cual el equipo de desarrollo cuenta con suficiente material de trabajo traducido en Historias de Usuario, que se recopilan en esta etapa para producir una primera entrega. También se produce el contacto necesario con las herramientas y tecnologías que se emplean para construir la aplicación y algunas ideas experimentales. Además, se pueden explorar soluciones puntuales cuando no se tenga una concepción transparente de alguna funcionalidad.

2.3.1 Historias de Usuario

Una de las mejores prácticas adoptadas en el desarrollo de software es la administración de requerimientos. XP propone hacer uso de las Historias de Usuario (HU) como técnica para especificar las funcionalidades que brinda el sistema y constituye una manera dinámica de realizar esta actividad. Como su nombre lo indica son especificadas por los propios usuarios y por tanto redactadas en su lenguaje; de manera sencilla y breve, evitando tecnicismos innecesarios que puedan crear confusión, aunque los programadores pueden contribuir en la tarea. Además, son la base para realizar las pruebas de aceptación, así como la estimación y planificación del proyecto. Una vez identificadas las historias de usuario necesarias para liberar una primera versión operativa los programadores proceden a descomponer cada una en tareas específicas, las denominadas tareas de programación que están escritas técnicamente y que darán solución a su Historia de Usuario correspondiente.

El resultado obtenido durante el trabajo realizado en la fase de Exploración permitió identificar las siguientes Historias de Usuarios.

Historia de Usuario		
Nombre: Solicitar nombre de la base de datos.	Puntos (Ptos.) Estimación: 1	
No.: 1	Usuario: Usuario del proyecto.	Iteración: 1
Descripción: El usuario debe introducir el nombre de la base de datos. El sistema muestra una interfaz que le permite al usuario introducir al sistema la información que se solicita.		
Prioridad: Alta.	Nivel de Complejidad: Baja.	
Información adicional:		

Tabla 1 HU Solicitar nombre de la base de datos

Historia de Usuario		
Nombre: Solicitar usuario y contraseña de la base de datos.		Puntos (Ptos.) Estimación: 1
No.: 2	Usuario: Usuario del proyecto.	Iteración: 1
Descripción: El usuario debe introducir la información solicitada por el sistema. El sistema muestra una interfaz con los campos requeridos donde el usuario es el encargado de escribir la información solicitada de la base de datos.		
Prioridad: Alta.		Nivel de Complejidad: Baja.
Información adicional:		

Tabla 2: HU Solicitar usuario y contraseña de la base de datos

Historia de Usuario		
Nombre: Localizar dirección del sistema eXcriba.		Puntos (Ptos.) Estimación: 1
No.: 3	Usuario: Usuario del proyecto.	Iteración: 1
Descripción: El sistema muestra una interfaz que le permite al usuario buscar la ubicación del eXcriba y posteriormente introducirla al sistema.		
Prioridad: Alta.		Nivel de Complejidad: Media.
Información adicional:		

Tabla 3: HU Localizar dirección del sistema eXcriba

Historia de Usuario		
Nombre: Elegir lugar para guardar la copia		Puntos (Ptos.) Estimación: 1
No.: 4	Usuario: Usuario del proyecto.	Iteración: 1
Descripción: El usuario selecciona el sitio donde desea guardar la copia. El sistema muestra una interfaz que le permita al usuario realizar la operación.		
Prioridad: Alta.		Nivel de Complejidad: Media.
Información adicional:		

Tabla 4: HU 4 Elegir lugar para guardar la copia

Historia de Usuario		
Nombre: Guardar configuración del sistema.		Puntos (Ptos.) Estimación: 1
No.: 5	Usuario: Usuario del proyecto.	Iteración: 1
Descripción: El usuario selecciona la opción "Guardar configuración". El sistema captura los datos introducidos por el usuario y almacena la configuración para posteriormente crear la copia cuando se ejecute la acción.		
Prioridad: Baja.		Nivel de Complejidad: Baja.
Información adicional:		

Tabla 5: HU 5 Guardar configuración del sistema

Historia de Usuario		
Nombre: Realizar copia de seguridad del sistema.		Puntos (Ptos.) Estimación: 2
No.: 6	Usuario: Usuario del proyecto.	Iteración: 2
Descripción: El usuario selecciona la opción "Realizar copia de seguridad". El sistema crea una copia de seguridad del sistema.		
Prioridad: Alta.		Nivel de Complejidad: Alta.
Información adicional:		

Tabla 6: HU 6 Realizar copia de seguridad del sistema

Historia de Usuario		
Nombre: Realizar restaura del sistema.		Puntos (Ptos.) Estimación: 2
No.: 7	Usuario: Usuario del proyecto.	Iteración: 2
Descripción: El usuario selecciona la opción "Realizar restaura". El sistema muestra el un listado de todas copias de seguridad realizadas anteriormente y restaura la información deseada por el usuario.		
Prioridad: Alta.		Nivel de Complejidad: Alta.
Información adicional: Tener en cuenta que el usuario debe seleccionar la copia de respaldo que desea restaurar.		

Tabla 7: HU 7 Realizar restaura del sistema

Prototipos de Interfaz de Usuario

En el presente acápite se muestran los prototipos de interfaz de usuarios en correspondencia con cada una de las Historias de Usuario.



Figura 2 *Página principal de la herramienta.*



Figura 3 *HU 1, 2, 3, 4.*



Figura 4 HU 5



Figura 5 HU 6



Figura 6 HU 7

Ver las restantes interfaces de usuario en los Anexos, figuras 13, 14, 15.

2.3.1 Requisitos No Funcionales

Los requisitos no funcionales son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo sobre el proceso de desarrollo y estándares. A menudo se aplican al sistema en su totalidad. Se aplican a características o servicios individuales del sistema. Los requerimientos no funcionales, como su nombre sugieren, son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste, como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. Definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema[40] Por lo general los requisitos no funcionales son fundamentales en el éxito del producto; normalmente están vinculados a los requisitos funcionales, es decir, una vez que se conoce lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades o propiedades debe tener.

Los requisitos no funcionales han de especificarse cuantitativamente, siempre que sea posible para que se pueda verificar su cumplimiento. A continuación se muestran los requisitos no funcionales que se han definido para el desarrollo de la herramienta por parte del equipo de trabajo.

Requisitos de software.

- Máquina Virtual de Java 7. **Requisitos de Hardware.**
- Procesador: Intel, 3.0 GHz.
- Memoria RAM: 2Gb o superior.
- Espacio libre en disco duro: 1GB o superior.

Restricciones del diseño y la implementación.

- Se hace uso de las herramientas GDA Xabal eXcriba versión 3.1, Duplycity y el IDE NetBeans.
- El lenguaje de programación que será usado para la implementación es Java en su versión 7, siguiendo el paradigma de la Programación Orientada a Objetos y Bash.

Requisitos de usabilidad.

1. Facilidad de uso por parte de los usuarios: La interfaz debe ser lo más descriptiva posible, permitiendo que las operaciones a realizar por los usuarios estén bien descritas, de manera que se puedan entender claramente.
2. La interfaz debe tener mensajes contextuales asociados a los objetos.

Requisitos de portabilidad.

- La herramienta para la creación y restauración de copias de respaldos una vez creada podrá ser utilizada solo sobre distribuciones basadas en Debian.

2.4 Planificación

A lo largo de esta fase los clientes establecen la prioridad de las Historias de Usuario de acuerdo a sus necesidades más inmediatas, para luego asignarlas por orden de relevancia a las iteraciones planificadas. A partir de las Historias de Usuario se realiza la estimación del esfuerzo que se requiere por parte del equipo para su posterior implementación. El método escogido para realizar la estimación tiene como elemento el punto, el cual equivale a una semana perfecta de trabajo, esto se refiere a que solamente el equipo se dedica a labores relacionadas con la construcción del sistema sin la influencia o el retraso provocado por otros factores, lo que en la práctica es complejo lograr.

2.4.1 Estimación de esfuerzos por Historias de Usuarios.

Las estimaciones del esfuerzo para implementar las Historias de Usuario permiten tener una medida bastante real de la velocidad de progreso del proyecto y brindan una guía razonable a la cual ajustarse. Los resultados obtenidos de las estimaciones se exhiben a continuación.

Estimación por Historias de Usuario.		
No.	Historias de Usuarios	Ptos. de Estimación.
1	Solicitar el nombre de la base de datos.	1
2	Solicitar usuario y contraseña de la base de datos.	1
3	Localizar dirección del sistema eXcriba.	1
4	Elegir lugar para guardar la copia del sistema.	1
5	Guardar configuración del sistema.	1
6	Realizar copia de seguridad del sistema.	2
7	Realizar restaura del sistema.	2

Tabla 8: Ptos. de Estimación por Historias de Usuario

2.4.2 Plan de Liberaciones

El Plan de Liberaciones (entregas) tiene como objetivo definir el número de entregas que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una. El cliente se encarga de decidir cuáles Historias de Usuario comprende la primera entrega según sus prioridades para darle valor a su negocio y que por tanto justifique su ejecución y así sucesivamente para las demás. Se decide utilizar un plan con fecha fija debido a las restricciones que impone el ambiente de producción. Para realizar el plan de entregas, el sistema propuesto se ha dividido en módulos que contienen las Historias de Usuario relacionadas lógicamente de acuerdo a su propósito. Cada una de las entregas se corresponde con una iteración de desarrollo.

Módulo	Historias de Usuario
Configuración	Solicitar el nombre de la base de datos.
	Solicitar usuario y contraseña de la base de datos.
	Localizar dirección del sistema eXcriba.
	Elegir lugar para guardar la copia del sistema.
	Guardar configuración del sistema.

Tabla 9: Historias de Usuario por módulos

Módulo	Historias de Usuario
Copia y Restaura	Realizar copia de seguridad del sistema.
	Realizar restaura del sistema.

Tabla 10: Historias de Usuario por módulos

El plan de liberaciones pronosticado queda finalmente estructurado en 2 iteraciones. El mismo consta de 7 Historias de Usuario; obteniéndose los resultados en forma de versiones operacionales, los cuales se exponen en la siguiente tabla.

Herramienta	Iteración 1	Iteración 2
Creación y restauración de copias de seguridad	21/02/2016	15/04/2016
	12/03/2016	06/05/2016
	v0.1	v1.0 final

Tabla 11: Plan de entregas

2.5 Plan de Iteraciones

En el plan de liberaciones se establece cuántas iteraciones serán necesarias realizar sobre el sistema para su éxito. Sin embargo, se precisa establecer el contenido de trabajo para todas y cada una de ellas y es aquí donde hace acto de presencia el plan de iteraciones, regulando la cantidad de Historias de Usuario a implementar dentro del rango establecido por la estimación efectuada. Tomando como referencia los aspectos antes tratados, el componente de salva y restaura que se pretende construir se desarrollará en 2 iteraciones, las cuales se detallan a continuación.

2.5.1 Iteración 1

La presente iteración tiene la finalidad de implementar las Historias de Usuario que se consideran más necesarias atendiendo a su relevancia e impacto para el negocio. Además, se establece una arquitectura para el software que puede ser utilizada durante toda la trayectoria evolutiva del resto del proyecto. También se dan respuestas a las funcionalidades: Solicitar nombre de la base de datos, Solicitar usuario y contraseña de la base de datos, Elegir lugar para guardar la copia, Localizar dirección del eXcriba.

2.5.2 Iteración 2

En esta iteración se dan cumplimiento a las Historias de Usuarios que no han sido implementadas en la iteración anterior. Se implementan las funcionalidades Guardar configuración del sistema, Realizar copia de seguridad del sistema y Realizar restaura del sistema. Finalizada esta iteración se obtendrá una versión con más de un 95 % de la implementación de la herramienta informática.

2.5.3 Plan de duración de las iteraciones.

XP es una de las metodologías que más enfatiza el uso de pruebas, proponiendo esta práctica como una de sus prioridades fundamentales para verificar la calidad; aplicándola desde el comienzo hasta la conclusión del software. De manera general dentro de cada iteración se ejecutan las pruebas unitarias y las de aceptación para tener la certeza de que se implementaron correctamente los requerimientos. Las nuevas funcionalidades que surgen con las otras funcionalidades ya elaboradas, también se chequean mediante las pruebas de integración. Se debe integrar continuamente, y de este modo se garantiza el seguimiento de cada una de las etapas. A modo de resumen se presenta la siguiente tabla que muestra las dos iteraciones analizadas previamente con las Historias de Usuario y su duración.

Iteraciones	Historias de Usuario a implementar	Duración
Iteración 1	Solicitar el nombre de la base de datos. Solicitar usuario y contraseña de la base de datos. Localizar dirección del sistema eXcriba. Elegir lugar para guardar la copia del sistema	3 semanas
Iteración 2	Guardar configuración del sistema. Realizar copia de seguridad del sistema. Realizar restaura del sistema.	3 semanas

Tabla 12: Plan de duración de las iteraciones

2.6 Conclusiones parciales

El presente capítulo permitió analizar los principales conceptos asociados a la gestión de copias de respaldo, pudiendo incluir que:

- La comprensión del funcionamiento de la gestión de copias locales en la herramienta eXcriba permitió definir las principales funcionalidades con que debe contar la herramienta informática a desarrollar.
- Se realizó una descripción de la solución propuesta para resolver el problema planteado teniendo en cuenta estas funcionalidades.

Una vez realizado este análisis se procede a efectuar el proceso de diseño de la herramienta informática para la creación y restauración de copias de seguridad.

Capítulo 3: Diseño, codificación y pruebas de la aplicación

En este capítulo se hace referencia a los elementos que conforman el diseño de la herramienta a construir. Se describen las pruebas de aceptación que se le aplicaron a la herramienta para verificar su correcto funcionamiento y que éste responda a las necesidades del proyecto.

3.1 Diseño de la herramienta informática

XP establece prácticas especializadas que inciden directamente en la realización del diseño para lograr un sistema robusto y reutilizable tratando de mantener su simplicidad, es decir, crear un diseño evolutivo que se va mejorando incrementalmente y que permite hacer entregas pequeñas y frecuentes de valor para el cliente. A la hora de darle cumplimiento a la actividad de diseñar, XP no especifica ninguna técnica de modelado, puede utilizarse indistintamente sencillos esquemas en una pizarra, diagramas de clases utilizando UML o tarjetas CRC (Clase, Responsabilidad y Colaboración) siempre que sean útiles, tributen a la comprensión y no requieran mucho tiempo en su creación.

3.1.1 Tarjetas CRC

Las tarjetas CRC (clase-responsabilidad-colaborador) identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual del proyecto [28]. Esta técnica de modelado permite entender las características de la aplicación pensando en términos de objetos y clases. Se utilizó la estrategia tormenta de ideas que representa una parte importante en el uso de las tarjetas y constituye el paso inicial del proceso. La plantilla de tarjeta CRC utilizada y los resultados obtenidos se describen a continuación.

Clase: Settings	
Descripción: es la encargada de almacenar los parámetros que se le solicitan al usuario para la configuración de la herramienta.	
Responsabilidad	Colaborador
Nombre Base de Datos	Execution
Usuario Base de Datos.	Execution
Contraseña Base de Datos.	Execution
Localizar eXcriba.	Execution
Lugar de destino del backup	Execution
Guardar configuración.	Execution

Tabla 13: Tarjeta CRC Settings

Clase: Execution	
Descripción: se encarga de ejecutar los comandos principales y establece conexión con el script para ejecutar sus funcionalidades.	
Responsabilidad	Colaborador
Realizar copia de seguridad.	Settings
Restaurar sistema.	Settings

Tabla 14: Tarjeta CRC Execution

3.1.2 Estructura de la herramienta informática

El proyecto se compone por varios paquetes con el objetivo de crear una organización entre las clases que lo integran. Los cuales son mostrados a continuación exponiendo su estructura.

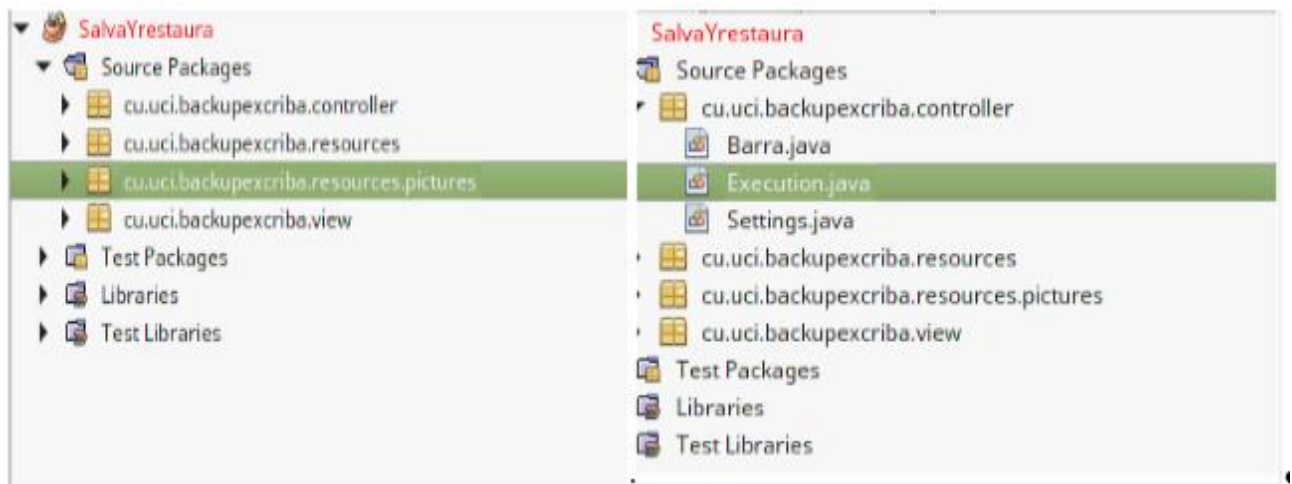


Figura 7 Estructura de la herramienta informática

Esta distribución permite organizar de forma entendible el código fuente de la aplicación, logrando disminuir el tiempo de desarrollo de la aplicación. Además, posibilita realizar el mantenimiento de la herramienta informática, así como la incorporación de nuevas funcionalidades.

- **cu.uci.backupexcriba.controller:** Consta de tres clases, donde la clase Execution es la encargada de ejecutar operaciones tales como: realizar copia de seguridad, guardar configuración, listar backups, realizar restaura del sistema. La clase Settings tiene definido todos los parámetros de la configuración de la herramienta. En la clase Bar es donde se implementa un hilo para mostrar la barra de progreso de realización del respaldo.
- **cu.uci.backupexcriba.resources:** Contiene el script que ejecuta los procesos de creación y restauración de copias, las propiedades y otros recursos.
- **cu.uci.backupexcriba.view:** Contiene todas las interfaces de usuario diseñadas para la herramienta informática.

- **cu.uci.backupexcriba.resources.pictures:** Contiene las imágenes utilizadas en la aplicación, tales como: los botones.

3.1.3 Arquitectura de la herramienta informática

Para el diseño de la arquitectura de la herramienta a desarrollar se ha combinado el estilo y patrón arquitectónico en capas, los cuales se explican a continuación.

Un estilo arquitectónico es una transformación que se impone al diseño de todo el sistema. El objetivo es establecer una estructura para todos los componentes del sistema. En el caso en el que ha de hacerse la reingeniería de una arquitectura ya existente, la imposición de un estilo arquitectónico dará como resultado cambios fundamentales en la estructura del software, incluida la reasignación de las funciones de los componentes[28].

Los patrones arquitectónicos se abocan a un problema de aplicación específica dentro de un contexto dado y sujeto a limitaciones y restricciones. El patrón propone una solución arquitectónica que sirve como base para el diseño de la arquitectura[28]. Se ha escogido el estilo y patrón de arquitectura en capas, el cual consiste en definir un número de capas diferentes; cada una ejecuta operaciones que se aproximan progresivamente al conjunto de instrucciones de máquina[28].

Para el diseño de la arquitectura de la nueva herramienta informática se definieron solo dos capas. La primera capa la constituye la Lógica de Negocio, pues la misma contiene todas las clases de datos únicos capturados por la herramienta a través de la interfaz. La segunda capa es la capa Presentación, la cual tiene las clases vistas construidas en swing y éstas son las encargadas de mostrarle al usuario una interfaz agradable y de guiarlo a la hora de trabajar con la aplicación. También tiene una carpeta denominada recursos que contiene las librerías, imágenes y otros recursos que se han utilizado para la implementación de dicha herramienta informática.

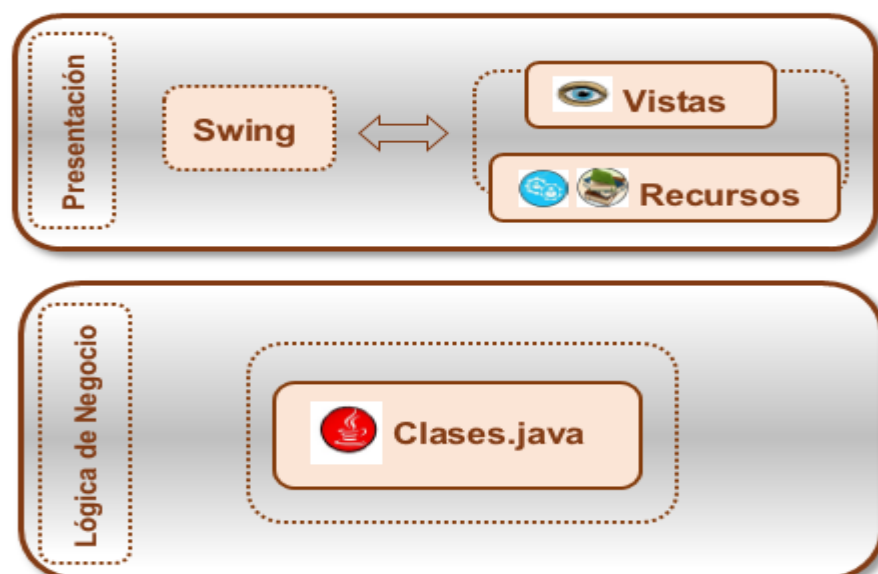


Figura 8 Arquitectura de la herramienta informática

3.1.5 Patrones de diseño

Un patrón de diseño describe una estructura de diseño que resuelve un problema particular del diseño dentro de un contexto específico y entre "fuerzas" que afectan la manera en la que se aplica y en la que se utiliza dicho patrón. Su objetivo es proporcionar una descripción que permita a un diseñador determinar

- Si el patrón es aplicable al trabajo en cuestión.
- Si puede volverse a usar (con lo que se ahorra tiempo de diseño).
- Si sirve como guía para desarrollar un patrón distinto en funciones o estructuras[28].

Para llevar a cabo la implementación de la herramienta informática, se utilizaron los patrones que a continuación se exponen.

- **Patrón Experto:** Es la clase que tiene la información necesaria para realizar la responsabilidad. Este se utiliza a la hora de modelar las entidades persistentes de la solución donde las funcionalidades propias de cada información re-implementan en la clase que contiene toda la información, o sea, la clase experta en la información[41].
- **Patrón Alta Cohesión:** Más que un diseño directamente en código, se trata de un principio que guiará el diseño del software. Plantea que la información almacenada en una clase debe de ser coherente y debe estar (en la medida de lo posible) relacionada con la clase[41].
- **Patrón Bajo Acoplamiento:** Da respuesta a la problemática de soportar bajas dependencias, bajo impacto del cambio e incremento de la reutilización. El acoplamiento es una medida de la fuerza con que un elemento está conectado a, tiene conocimiento de, confía en, otros elementos[41].

3.2 Codificación de la herramienta informática

En la fase inicial de Exploración se identificaron y detallaron las Historias de Usuario. Posteriormente en la fase de Planificación, el cliente seleccionó en correspondencia con sus prioridades y restricciones de tiempo, qué construir. Una vez realizada estas actividades, el equipo de desarrolladores entra en escena para darle cumplimiento a lo acordado. Las historias de usuario elegidas se descomponen en tareas de programación o ingeniería, escritas en lenguaje técnico, que a su vez son convertidas posteriormente a código. La programación en parejas, la refactorización, la integración continua y la compilación de diez minutos son un subconjunto de las prácticas que guían esta crucial actividad.

3.2.1 Iteración 1

Se desarrolla la Historia de Usuario número uno, la cual permite introducir el nombre de la base de datos a la herramienta por parte del usuario.

Tarea	
Número de la tarea: 1	Número de la Historia de Usuario: 1
Nombre de la tarea: Diseño de la interfaz Nombre Base de Datos.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 21/02/2016	Fecha fin: 12/03/2016
Programador responsable: Mirna Despaigne Biset, Glauver Matos Fonseca.	
Descripción: Muestra un área de texto, la cual le permite al usuario escribir el nombre de la base de datos.	

Tabla 15: HU 1 Solicitar el nombre de la base de datos

Se desarrolla la Historia de Usuario número dos, la cual permite introducir el usuario de la base de datos a la herramienta.

Tarea	
Número de la tarea: 2	Número de la Historia de Usuario: 2
Nombre de la tarea: Diseño de la interfaz Usuario Base de Datos.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 21/02/2016	Fecha fin: 12/03/2016
Programador responsable: Mirna Despaigne Biset, Glauver Matos Fonseca.	
Descripción: Muestra un área de texto, la cual le permite al usuario escribir el usuario de la base de datos.	

Tabla 16: HU 2 Solicitar usuario y contraseña de la base de datos

Se desarrolla la Historia de Usuario número dos, la cual permite introducir la contraseña de la base de datos a la herramienta por parte del usuario.

Tarea	
Número de la tarea: 3	Número de la Historia de Usuario: 2
Nombre de la tarea: Diseño de la interfaz Contraseña Base de Datos.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 21/02/2016	Fecha fin: 12/03/2016
Programador responsable: Mirna Despaigne Biset, Glauver Matos Fonseca.	
Descripción: Muestra un área de texto, la cual le permite al usuario escribir la contraseña de la base de datos.	

Tabla 17: HU 2 Solicitar usuario y contraseña de la base de datos

Se desarrolla la Historia de Usuario número tres, la cual permite buscar la localización del sistema eXcriba e introducirlo a la configuración de la herramienta.

Tarea	
Número de la tarea: 4	Número de la Historia de Usuario: 3
Nombre de la tarea: Diseño de la interfaz Localizar eXcriba.	
Tipo de tarea: Desarrollo.	Puntos estimados:
Fecha inicio: 21/02/2016	Fecha fin: 12/03/2016
Programador responsable: Glauver Matos Fonseca, Mirna Despaigne Biset.	
Descripción: Muestra un botón que le permite al usuario buscar la ubicación actual del sistema eXcriba. Al lado de este botón se encuentra un campo de texto que captura la dirección seleccionada por el usuario.	

Tabla 18: HU3 Localizar dirección del sistema eXcriba

Se desarrolla la Historia de Usuario número cuatro, la cual permite buscar el destino que se del sistema eXcriba e introducirlo a la configuración de la herramienta.

Tarea	
Número de la tarea: 5	Número de la Historia de Usuario: 4
Nombre de la tarea: Diseño de la interfaz Lugar destino del backup.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 21/02/2016	Fecha fin: 12/03/2016
Programador responsable: Glauver Matos Fonseca, Mirna Despaigne Biset.	
Descripción: Muestra un botón que le permite al usuario buscar el lugar de destino de la copia. Al lado de este botón se encuentra un campo de texto que captura la dirección seleccionada por el usuario para guardar la copia de respaldo.	

Tabla 19: HU 4 Elegir lugar para guardar la copia del sistema

3.2.2 Iteración 2

Se desarrolla la Historia de Usuario número cinco, la cual permite guardar la configuración introducida por el usuario.

Tarea	
Número de la tarea: 6	Número de la Historia de Usuario: 5
Nombre de la tarea: Diseño de la interfaz Guardar configuración.	
Tipo de tarea: Desarrollo.	Puntos estimados: 1
Fecha inicio: 15/04/2016	Fecha fin: 06/05/2016
Programador responsable: Mirna Despaigne Biset, Glauver Matos Fonseca.	
Descripción: Muestra un botón que es el encargado de realizar la conexión con la clase Setting y guardar los parámetros de configuración.	

Tabla 20: HU 5 Guardar configuración

Se desarrolla la Historia de Usuario número seis, la cual permite crear una copia de seguridad del sistema eXcriba.

Tarea	
Número de la tarea: 7	Número de la Historia de Usuario: 6
Nombre de la tarea: Diseño de la interfaz Realizar copia de seguridad.	
Tipo de tarea: Desarrollo.	Puntos estimados: 2
Fecha inicio: 15/04/2016	Fecha fin: 06/05/2016
Programador responsable: Glauver Matos Fonseca, Mirna Despaigne Biset.	
Descripción: Muestra un botón que es el encargado de realizar la conexión con la clase Execution y realizar el proceso de creación de la copia de seguridad del sistema.	

Tabla 21: HU 6 Realizar copia de seguridad del sistema

Se desarrolla la Historia de Usuario número siete, la cual permite seleccionar una copia de seguridad del listado mostrado y realizar la restauración del respaldo seleccionado.

Tarea	
Número de la tarea: 8	Número de la Historia de Usuario: 7
Nombre de la tarea: Diseño de la interfaz Restaurar sistema.	
Tipo de tarea: Desarrollo.	Puntos estimados: 2
Fecha inicio: 15/04/2016	Fecha fin: 06/05/2016
Programador responsable: Glauver Matos Fonseca, Mirna Despaigne Biset.	
Descripción: Muestra un botón que es el encargado de realizar la conexión con la clase Execution y realizar el proceso de restauración del sistema.	

Tabla 22: HU 7 Realizar restaura del sistema

Para un mayor entendimiento de las funcionalidades implementadas, en las siguientes ilustraciones se muestran los métodos principales para la realización de la copia de seguridad y la restauración del sistema. Conjuntamente se realiza una breve descripción de los métodos.

Se realiza el llamado al método comando_backup perteneciente a la clase Execution, donde se conecta con el script en modo backup y es el script el encargado de ejecutar la salva del sistema.

```

*/
public boolean comando_backup(){
    if (_comando_backup("cs") && _comando_backup("db")){
        finishbackup=true;
        return true;
    } else
        return false;
}
private boolean _comando_backup(String tipo){
    try {
        int b;
        String args= resourcePath + "excribabackup.sh backup "+tipo;
        args = this.runas + args;
        Process process = Runtime.getRuntime().exec(args);
        InputStream inputstream = process.getInputStream();
        BufferedInputStream buffer = new BufferedInputStream(inputstream);
        while ( ( b = buffer.read() ) != -1 )
        {

            char c = (char)b;

            System.err.print(""+(char)b); //This prints out content that is unreadable.
            //Isn't it supposed to print out html tag?
            this.outputHistory +=(char)b;
        }
    }
    try {

```

Figura 9 Método para crear copias de seguridad

Se ejecuta el método comando_restore perteneciente a la clase Execution, donde se realiza la conexión con el script en modo restaura con la fecha de la salva correspondiente y el lugar de destino para la misma; donde el script es el encargado de realizar la restaura del sistema.

```

public boolean comando_restore(String fecha){
    if (_comando_restore("cs",fecha) && _comando_restore("db",fecha)){
        finishrestore=true;
        return true;
    }
    return false;
}
public boolean _comando_restore(String tipo, String fecha){
    try {
        int b;
        String args= resourcePath + "excribabackup.sh restore "+tipo+" "+fecha+" "+excribaPath+"/RESTORE";
        System.out.println(args);
        args = this.runas + args;
        Process process = Runtime.getRuntime().exec(args);
        InputStream inputstream = process.getInputStream();
        BufferedInputStream buffer = new BufferedInputStream(inputstream);
        while ( ( b = buffer.read() ) != -1 )
        {

            char c = (char)b;

            System.err.print(""+(char)b); //This prints out content that is unreadable.
            //Isn't it supposed to print out html tag?
            this.outputHistory +=(char)b;
        }
    }
    try {

```

Figura 10 Método para la restauración de copias de seguridad

Cuando se selecciona la copia de respaldo que se desea restaurar mostrada en el Jtable de la clase visual Principal, se devuelve del Jtable la fecha en formato AA-MM-DD en la cual se va a realizar la restaura. Ver anexo, figura 16.

Cuando se selecciona la opción Realizar restaura del sistema, automáticamente se ejecuta el método `get_restore_list` que ejecuta el script para listar los backup actualmente contenidos en la carpeta de destino. Ver anexo, figura 17.

3.3 Pruebas sobre la herramienta informática

Las pruebas constituyen una parte fundamental en la filosofía XP, que no se debe obviar bajo ningún concepto y que debe aplicarse de manera frecuente y temprana. Varias de las principales prácticas que sugiere se apoyan en este enfoque como vía para obtener resultados satisfactorios. La actividad de probar el sistema y verificar que se encuentra libre de defectos, tiene muchos beneficios. Por ejemplo, la calidad es una variable muy importante para todo producto y uno de los caminos para garantizarla es siguiendo esta doctrina. La confianza en el equipo también se apoya sobre esta sólida base. Por último, proporciona una medida del progreso del trabajo que despliega el equipo.

3.3.1 Pruebas unitarias

La prueba de unidad enfoca los esfuerzos de verificación en la unidad más pequeña del diseño del software: el componente o módulo del software. Al usar la descripción de diseño del componente como guía, las rutas de control importantes se prueban para descubrir errores dentro de la frontera del módulo. Las pruebas de unidad se enfocan en la lógica del procesamiento interno y de las estructuras de datos dentro de las fronteras de un componente. Por lo general, se consideran como adjuntas al paso de codificación[28].

Para realizar las pruebas unitarias a la herramienta se utilizó el framework JUnit, el cuál es un framework de pruebas gratuito en el que se define un conjunto amplio de clases que automatizan la ejecución de pruebas unitarias para software orientado a objetos, en particular de programas Java[42]. Está basado en función de algún valor de entrada, evalúa el valor de retorno esperado y si la clase cumple con la especificación devuelve que pasó exitosamente la prueba; en caso contrario, devolverá un fallo en el método correspondiente. Es también un medio de controlar los nuevos resultados cuando una parte del código ha sido modificado y se desea saber si el nuevo código cumple con los requerimientos anteriores y que no se ha alterado su funcionalidad después de la nueva modificación.

El framework se utilizó para probar los resultados de las funcionalidades implementadas en la clase Execution. A continuación, se muestra una imagen del resultado obtenido en la realización de la prueba unitaria al método Configurar.

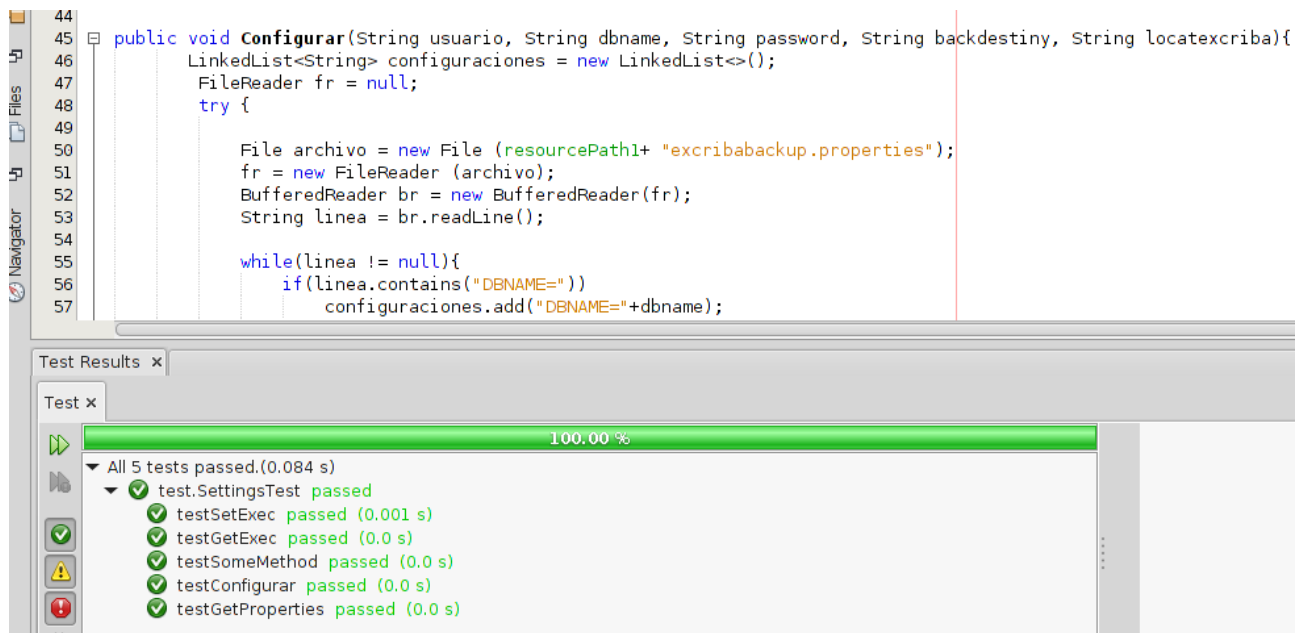


Figura 11 Resultado de la prueba unitaria al método Configurar en JUnit

Para más información sobre las pruebas unitarias realizadas ver Anexos figura 18, 19.

3.3.3 Pruebas de integración

Las pruebas de integración son una técnica sistemática para construir la arquitectura del software mientras lleva a cabo pruebas para descubrir errores asociados con la interfaz. El objetivo es tomar los componentes probados de manera individual y construir una estructura de programa que se haya dictado por diseño[28]. Una vez que fueron acopladas todas las funcionalidades y componentes de la herramienta informática desarrollada, se le aplicaron casos de pruebas funcionales correspondientes para verificar su correcto funcionamiento. Los errores detectados en el funcionamiento de la misma fueron resueltos, obteniéndose posteriormente resultados satisfactorios en su totalidad.

3.3.4 Pruebas de aceptación

Las pruebas de aceptación se crean a partir de historias de usuario. Durante una iteración las historias de usuario se seleccionan durante la reunión de planificación se traducirá en pruebas de aceptación de la iteración. El cliente especifica escenarios para probar cuando una historia de usuario se ha implementado correctamente. Las pruebas de aceptación son pruebas de caja negra del sistema. Cada prueba de aceptación representa un resultado esperado del sistema. Los clientes son responsables de verificar la corrección de las pruebas de aceptación y revisión de resultados

de las pruebas para decidir que pruebas fallidas son de máxima prioridad. También se utilizan como pruebas de regresión antes de un lanzamiento de producción[43]. A continuación, se muestran los casos de pruebas definidos para efectuar dicha prueba a la herramienta informática.

Caso de prueba para la historia de usuario: Guardar configuración del sistema.

Esta sección cubre el conjunto de pruebas que se le realizaron a la historia de usuario: Guardar configuración del sistema. En la cual se intenta capturar los datos introducidos por el usuario para configurar la herramienta.

Caso de prueba	
Identificador: HU5-PF1	Historia de Usuario: 5
Nombre: Guardar configuración del sistema.	
Descripción: Prueba para la funcionalidad de guardar la configuración del sistema.	
Condiciones de ejecución: Se deben capturar los datos de configuración del sistema.	
Entrada/Pasos de ejecución: El usuario introduce los datos y selecciona la opción Guardar configuración.	
Resultado esperado: Se deben guardar los datos de configuración del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 23: Caso de prueba Guardar configuración del sistema

Caso de prueba para la historia de usuario: Realizar copia de seguridad del sistema.

Esta sección cubre el conjunto de pruebas que se le realizaron a la historia de usuario: Realizar copia de seguridad del sistema. En la misma se intenta crear una copia de respaldo para el sistema GDA Xabal eXcriba 3.1.

Caso de prueba	
Identificador: HU6-PF2	Historia de Usuario: 6
Nombre: Realizar copia de seguridad del sistema.	
Descripción: Prueba para la funcionalidad de crear una copia de seguridad del sistema.	
Condiciones de ejecución: Se debe crear una copia de respaldo del sistema.	
Entrada/Pasos de ejecución: Luego de guardarse los datos de configuración del sistema. El usuario selecciona la opción Realizar copia de seguridad	
Resultado esperado: Se debe crear una copia de seguridad del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 24: Caso de prueba Realizar copia de seguridad del sistema

Caso de prueba para la historia de usuario: Realizar restaura del sistema.

Esta sección cubre el conjunto de pruebas que se le realizaron a la historia de usuario: Realizar restaura del sistema. En la misma se intenta realizar la restauración de una copia de respaldo para el sistema GDA Xabal eXcriba 3.1

Caso de prueba	
Identificador: HU7-PF3.1	Historia de Usuario: 7
Nombre: Realizar restaura del sistema.	
Descripción: Prueba para la funcionalidad de restaurar una copia de seguridad del sistema.	
Condiciones de ejecución: Se debe restaurar la copia de seguridad del sistema.	
Entrada/Pasos de ejecución: El usuario elige la copia de seguridad que desea restaurar. A continuación selecciona la opción Realizar restaura del sistema	
Resultado esperado: Se debe realizar la restauración de la copia de seguridad seleccionada del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 25: Caso de prueba Realizar restaura del sistema

3.5 Resultados de las pruebas

A continuación, se muestra gráficamente un resumen de las no conformidades detectadas en la herramienta informática, utilizando los casos de prueba diseñados:

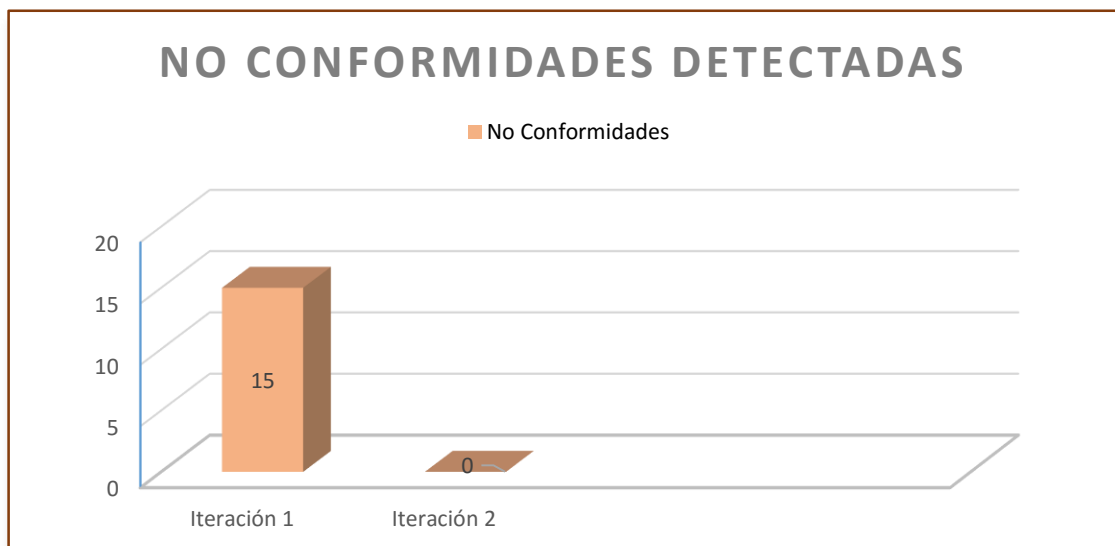


Figura 12 Resultados de las pruebas

Entre las principales no conformidades encontradas se encuentran:

- Faltas ortográficas (tildes) en algunas interfaces.
- Nombres incorrectos en diferentes interfaces.
- Errores de diseño en diferentes interfaces.
- Incorrecta validación de los formatos de datos de cada entrada efectuada por el usuario.
- Errores de implementación de algunas funcionalidades.

Las no conformidades detectadas durante las dos iteraciones realizadas fueron resueltas satisfactoriamente por el equipo de desarrollo.

3.6 Conclusiones parciales

En el presente capítulo se llevó a cabo la implementación de la herramienta informática para la creación y restauración de copias de seguridad, guiado por la metodología de desarrollo XP. Dicha herramienta informática garantiza la seguridad e integridad de los datos en el sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1. El código implementado estuvo enfocado en el cumplimiento de los requisitos funcionales y no funcionales antes descritos en las historias de usuario. El desarrollo de la misma y su desempeño en la ejecución de los procesos de creación y restauración de copias de respaldos, es la muestra de los resultados satisfactorios obtenidos en las etapas de diseño, codificación y pruebas.

Conclusiones generales

El presente trabajo describe el proceso de desarrollo de la herramienta informática para la creación y restauración de copias de seguridad para el sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1, siendo esta la respuesta al problema planteado. Para lo cual, se realizó un estudio del marco teórico en el que se sustenta la investigación, enfocado en las características que debe poseer la aplicación a desarrollar.

Para materializar la propuesta de solución se generó el artefacto Tarjeta CRC, basado en la metodología utilizada XP, orientado a elaborar el diseño e implementación de la herramienta informática. El artefacto generado, la implementación de la herramienta informática y su desempeño son la prueba del cumplimiento del objetivo general de la presente investigación: desarrollar una herramienta informática para la creación y restauración de copias de seguridad en el Gestor de Documentos Administrativos Xabal eXcriba 3.1, que garantice la seguridad e integridad de los datos en el sistema.

Como elementos concluyentes de la presente investigación se tiene:

- Se realizó un análisis del marco teórico en el que se sustenta la investigación, enfatizado en los procesos de creación y restauración de respaldos y las aplicaciones gestoras de copias de seguridad que facilitan la realización de dichas tareas.
- La etapa de diseño se realizó satisfactoriamente, definiéndose una arquitectura en capas para la nueva herramienta informática y obteniéndose los elementos necesarios para llevar a cabo la implementación de la misma, basada en la metodología XP.
- Con el desarrollo de la nueva herramienta informática se garantiza la seguridad e integridad de los datos del sistema GDA Xabal eXcriba 3.1, a través de la creación y restauración de copias de seguridad.
- Las pruebas realizadas a la aplicación validaron la correcta implementación de las funcionalidades descritas en las historias de usuario, demostrando un buen desempeño de la ejecución en los procesos de creación y restauración de copias de seguridad para el sistema GDA Xabal eXcriba 3.1.

Recomendaciones

- Integrar la herramienta informática para la creación y restauración de copias de seguridad al sistema Gestor de Documentos Administrativos Xabal eXcriba 3.1 con el objetivo de realizar las copias de seguridad directamente en el sistema.
- Implementar una nueva funcionalidad que permita definir la periodicidad con que se realizarán las copias de seguridad.
- Agregar la funcionalidad que permita la realización de copias y su restauración a través de localizaciones FTP.

Referencias Bibliográficas

- [1] Y. SANTANA, *Excriba como solución para la Gestión Documental*. Ciudad de La Habana.: Ediciones Futuro, 2011.
- [2] R. ELEJALDE, *Módulo para la creación de modelos de contenido para Alfresco*. Universidad de las Ciencias Informáticas. Ciudad de la Habana, Cuba., 2009.
- [3] "Recovery Labs. Principales factores que causan una pérdida de información." [Online]. Available: <http://www.recoverylabs.com/ayuda-y-soporte/data-recovery-white-papers/informes/principales-factores-que-causan-una-perdida-de-informacion/>.
- [4] P. W. Curtis, *Backup and Recovery*. 2006.
- [5] "Diccionario de Informática y Tecnología. Definición de backup (informática)." [Online]. Available: <http://www.alegsa.com.ar/Dic/backup.php>.
- [6] "Swedorski, Scoot. Handy backup the power of simplicity." [Online]. Available: <http://www.handybackup.es/>.
- [7] "Search Sotorage. PROVEEDORES DE TECNOLOGÍA DE ALMACENAMIENTO 2010." [Online]. Available: <http://www.searchstorage.es/acerca-de-nosotros/>.
- [8] "Osmosis Latina. Respaldos ('Backups and Redundancy')." [Online]. Available: <http://www.osmosislatina.com/soporte/respaldos.htm>.
- [9] "Document Management System Features _ eFileCabinet." [Online]. Available: <http://www.efilecabinet.com/document-management-system/>.
- [10] P. Hampton, "Alfresco One New Release Favorite Features _ Alfresco Blog," 1 noviembre, 2013. [Online]. Available: <https://www.alfresco.com/blogs/alfresco-one-new-release-features/>.
- [11] "docSTAR eclipse Review." [Online]. Available: <https://www.betterbuys.com/dms/reviews/docstar-eclipse/>.
- [12] D. M. W. Papers, "eclipse Cloud Document Management Software Features - docSTAR." [Online]. Available: <http://www.docstar.com/document-management-systems/eclipse/features>.
- [13] "Document Management Software _ Ademero." .
- [14] "Best Compares. Features of Content Central Document Management Software by Ademero." [Online]. Available: <http://document-management.bestcompares.com/d/content-central-6.html>.
- [15] "Content Central by Ademero Software - 2016 Reviews, Pricing & Demo," 2016. [Online]. Available: <http://www.softwareadvice.com/cms/content-central-ademero-profile/>.
- [16] "SmartFile _ File Sharing Solutions for Business & Enterprise." [Online]. Available: <http://www.smartfile.com/blog/>.
- [17] "Reviews of SmartFile Free Pricing & Demos Document Management Software." [Online]. Available: <http://www.capterra.com/document-management->

- software/spotlight/134146/SmartFile/SmartFile.
- [18] “Características y funcionalidades de Athento iDM _ Athento.” [Online]. Available: <http://www.athento.com/caracteristicas-tecnicas-athento-iDM/>.
- [19] “Portal Programas. Knowledgetree.” [Online]. Available: <http://www.portalprogramas.com/knowledgetree/>.
- [20] “Copias de seguridad con rsync.” [Online]. Available: <http://www.horus.es/ss/blog/copias-de-seguridad-con-rsync/>.
- [21] “Duplicity - Linux Links - The Linux Portal Site.” [Online]. Available: <http://www.linuxlinks.com/article/20090106115114432/Duplicity>.
- [22] “luckyBackup - backup and sync utility.” [Online]. Available: <https://mislinuxapps.wordpress.com/?s=luckybackup>.
- [23] “luckyBackup - features.” [Online]. Available: <http://luckybackup.sourceforge.net/manual.html>.
- [24] “LuckyBackup, tus backups nunca fueron tan simples.” [Online]. Available: <http://ubunlog.com/luckybackup-tus-backups-nunca-fueron-tan-simples/>.
- [25] “dump_restore.” [Online]. Available: <http://ibiblio.org/pub/Linux/docs/LuCaS/Manuales-LuCAS/doc-unixsec/unixsec-html/node104.html>.
- [26] R. Barzanallana., “Apuntes Informática Aplicada a la Gestión Pública. Capítulo 2,” in *Apuntes Informática Aplicada a la Gestión Pública.*, 2012.
- [27] P. Letelier and C. Penadés, “Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP),” vol. 05, no. 1666–1680, 2006.
- [28] R. S. Pressman and D. Ph, *Ingeniería del software. Un enfoque práctico.*, 7ma ed. 2010.
- [29] L. Calabria and P. Píriz, *Metodología XP. Cátedra de Ingeniería de Software*. Uruguay: Universidad ORT Uruguay, 2003.
- [30] “DICCIONARIO DE INFORMÁTICA Y TECNOLOGÍA. Definición de Lenguaje de programación -.” [Online]. Available: [http://www.alegsa.com.ar/Dic/lenguaje de programacion.php](http://www.alegsa.com.ar/Dic/lenguaje_de_programacion.php).
- [31] “Software Terms. Java Definition,” 19 abril, 2012. [Online]. Available: <http://techterms.com/definition/java>.
- [32] J. García, J. I. Rodríguez, I. Mingo, and A. Imaz, *Aprenda Java como si estuviera en primero*. San Sebastián.: Universidad de Navarra, 2000.
- [33] “IBM Bluemix. Aprenda Linux, 101 La línea de comando Linux.” [Online]. Available: <http://www.ibm.com/developerworks/ssa/linux/library/l-lpic1-v3-103-1/>.
- [34] “What is integrated development environment (IDE)? - Definition from WhatIs.com.” [Online]. Available: <http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>.
- [35] “Bienvenido a NetBeans y www.netbeans.org, Portal del IDE Java de Código Abierto.” [Online]. Available: https://netbeans.org/index_es.

- [36] “Sublime Text ¿el editor de texto definitivo?” [Online]. Available:
<http://www.genbetadev.com/herramientas/sublime-text-2-el-editor-de-texto-definitivo>.
- [37] “Herramientas Case.” [Online]. Available: <http://fds-herramientascase.blogspot.com>.
- [38] “Software Design Tools for Agile Teams, with UML, BPMN and More.” [Online]. Available:
<http://www.visual-paradigm.com/>.
- [39] “Ingeniería del Software.” [Online]. Available:
<http://www.is.ls.fi.upm.es/docencia/is2/documentacion/ModeloDominio>.
- [40] “Bligoo. Requerimientos Funcionales y No Funcionales.” [Online]. Available:
<http://ingenieriadesoftware.bligoo.com.mx/requerimientos-funcionales-y-no-funcionales-rf-rnf>.
- [41] C. Larman, *UML y Patrones. Introducción al análisis y diseño orientado a objeto.*, 2da ed. México : Prentice Hall, 2003, 2003.
- [42] M. Polo, “Escuela Superior de Informática.” 2010. [Online]. Available: <http://www.inf-cr.uclm.es/www/mpolo/tutorial/proposito.html>.
- [43] “Acceptance test.” [Online]. Available:
<http://www.extremeprogramming.org/rules/functionaltests.html>.

Bibliografías

1. Acceptance test. (n.d.). Retrieved from <http://www.extremeprogramming.org/rules/functionaltests>
2. Adictos al trabajo. Tipos de backups. (n.d.). Retrieved from <https://www.adictosaltrabajo.com>
3. Backup Definition. (2008). Retrieved from <http://techterms.com/definition/backup>
4. Barzanallana., R. (2012). Apuntes Informática Aplicada a la Gestión Pública. Capítulo 2. In Apuntes Informática Aplicada a la Gestión Pública. Retrieved from <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo>
5. Bash, D. E. L. (n.d.). Comandos shells y programación en la shell del bash.
6. Best Compares. Features of Content Central Document Management Software by Ademero. (n.d.). Retrieved from <http://document-management.bestcompares.com/d/content-central-6.html>
7. Best Document Management Software _ 2016 Reviews of the Most Popular Systems. (n.d.). Retrieved from <http://www.capterra.com/document-management-software/>
8. Bienvenido a NetBeans y www.netbeans.org, Portal del IDE Java de Código Abierto. (n.d.). Retrieved from https://netbeans.org/index_es
9. Bligoo. Requerimientos Funcionales y No Funcionales. (n.d.). Retrieved from <http://ingenieriadesoftware.bligoo.com.mx/requerimientos-funcionales-y-no-funcionales-rf-rnf>
10. Bussines Dictionary. What is file backup definition and meaning. (n.d.).
11. Calabria, L., & Píriz, P. (2003). Metodología XP. Cátedra de Ingeniería de Software. Uruguay: Universidad ORT Uruguay.
12. Camacho, J. J. P. (2013). Administración de sistemas y servicios telemáticos. La herramienta dump. Backup de sistemas.
13. Características y funcionalidades de Athento iDM _ Athento. (n.d.). Retrieved from <http://www.athento.com/caracteristicas-tecnicas-athento-iDM/>
14. Católica, U., & San, B. (n.d.). Procedimientos de almacenamiento y respaldos. Elaboración de backups., 1–8.
15. Chapman, I. A. H., Chacón, I. R. E., & Samón, I. P. R. (2015). SOLUTION PROPOSAL TO THE BACKUPS MANAGEMENT WITHIN THE.
16. Coello González, S., & Hernández León, R. A. (2002). EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA. Ciudad de La Habana.: EDUNIV.
17. COLOMBIA, U. N. DE, ECONOMICAS, F. D. C., & COMUNICACIONES, U. I. Y. (n.d.). Guía de Backups o Copias de Seguridad.
18. Cómo realizar copias de seguridad con rsync. (n.d.). Retrieved from <http://www.eslomas.com/2005/01/como-realizar-copias-de-seguridad-con-rsync/>

19. Conekia labs. Copia de seguridad en Linux. (2012). Retrieved from <http://labs.conekia.es/copia-de-seguridad-en-linux-duplicity/>
20. Content Central by Ademero Software - 2016 Reviews, Pricing & Demo. (2016). Retrieved from <http://www.softwareadvice.com/cms/content-central-ademero-profile/>
21. Copias de seguridad con rsync. (n.d.). Retrieved from <http://www.horus.es/ss/blog/copias-de-seguridad-con-rsync/>
22. Cubanás, O., & Febles-rodríguez, M. M. J. P. (2010). LA GESTIÓN DOCUMENTAL COMO SOPORTE DEL SISTEMA DE GESTIÓN DEL CONOCIMIENTO EN LAS ORGANIZACIONES CUBANAS., 12 No. 1(15623297). Retrieved from http://www.ciget.pinar.cu/Revista/No.2010-1/Articulos/Gest_Documental.pdf
23. Curtis, P. W. (2006). Backup and Recovery.
24. Developer Network. Tipos de copia de seguridad. (2005). Retrieved from [https://msdn.microsoft.com/es-es/library/cc784306\(v=ws.10\).aspx](https://msdn.microsoft.com/es-es/library/cc784306(v=ws.10).aspx)
25. Diccionario de Informática y Tecnología. Definición de backup (informática). (n.d.). Retrieved from <http://www.alegsa.com.ar/Dic/backup.php>
26. DICCIONARIO DE INFORMÁTICA Y TECNOLOGÍA. Definición de Lenguaje de programación -. (n.d.). Retrieved from [http://www.alegsa.com.ar/Dic/lenguaje de programacion.php](http://www.alegsa.com.ar/Dic/lenguaje_de_programacion.php)
27. docSTAR eclipse Review. (n.d.). Retrieved from <https://www.betterbuys.com/dms/reviews/docstar-eclipse/>
28. Document Management Software _ Ademero. (n.d.). Retrieved from https://www.ademero.com/document_management_software/
29. Document Management System Features _ eFileCabinet. (n.d.). Retrieved from <http://www.efilecabinet.com/document-management-system/>
30. dump_restore. (n.d.). Retrieved from <http://ibiblio.org/pub/Linux/docs/LuCaS/Manuales-LuCAS/doc-unixsec/unixsec-html/node104.html>
31. Duplicity - Linux Links - The Linux Portal Site. (n.d.-a). Retrieved from <http://www.linuxlinks.com/article/20090106115114432/Duplicity>
32. Duplicity - Linux Links - The Linux Portal Site. (n.d.-b). Retrieved from <http://www.linuxlinks.com/article/20090106115114432/Duplicity>
33. Duplycity. (n.d.-a). Retrieved from <http://zonalinux.com.ar/como-configurar-a-nautilus-y-algunos-trucos/>.
34. Duplycity. (n.d.-b). Retrieved from <http://opensuse-tutorials.com/>
35. El shell Bash. (n.d.).
36. ELEJALDE, R. (2009). Módulo para la creación de modelos de contenido para Alfresco. Universidad de las Ciencias Informáticas. Ciudad de la Habana, Cuba.
37. Estrategias de backup. (n.d.), 286–291.
38. Fuente, T. D. la. (2008). Alfresco:gestión documental y gestión de contenidos web.

39. gadmin-rsync.softonic. (n.d.). Retrieved from <http://gadmin-rsync.softonic.com/linux/comparar/clonezilla,backharddi,synkron>
40. García, J., Rodríguez, J. I., Mingo, I., & Imaz, A. (2000). *Aprenda Java como si estuviera en primero*. San Sebastián.: Universidad de Navarra.
41. García, M. (n.d.). *Estrategias para realizar Copias de Seguridad y Restauración de Archivos*. Retrieved from <http://www.nettix.com.pe/manuales/administracion/288-estrategias-para-realizar-copias-de-seguridad-y-restauracion-de-archivos>
42. García, P. (n.d.). *Backup con Duplycity*. Retrieved from <http://pablogg.com/backup-con-duplicity/>
43. Gualtiero. (2014). *Tutorial: how to create a JUnit test method template in NetBeans*. Retrieved from <https://gualtierotesta.wordpress.com/2014/03/09/tutorial-how-to-create-a-junit-test-method-template-in-netbeans/>
44. Hampton, P. (2013). *Alfresco One New Release Favorite Features _ Alfresco Blog*. Retrieved from <https://www.alfresco.com/blogs/alfresco-one-new-release-features/>
45. Herramientas Case. (n.d.). Retrieved from <http://fds-herramientascase.blogspot.com>
46. IBM Bluemix. *Aprenda Linux, 101 La línea de comando Linux*. (n.d.). Retrieved from <http://www.ibm.com/developerworks/ssa/linux/library/l-lpic1-v3-103-1/>
47. Importancia de los "Backups." (2011). Retrieved from <http://www.zonavirus.com/articulos/importancia-de-los-backups.asp>
48. Ingeniería del Software. (n.d.). Retrieved from <http://www.is.ls.fi.upm.es/docencia/is2/documentacion/ModeloDominio>
49. Introducción de Rubén Venegas en Prezi. (n.d.). Retrieved from <http://prezi.com/pw8liegicqzi/introduccion/>
50. iSeries, IBM @ server. s.l. : Copyright International Business Machines Corporation 1998, 2001. (n.d.). *Planificación de una estrategia de copia de seguridad y recuperación*.
51. Jerez., A. G. (n.d.). *Backup manager: Herramienta de backup en sistema linux*. Retrieved from <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=backupmanager>
52. KnowledgeTree. *KnowledgeTree Backup And Disaster Recovery Information KnowledgeTree*. (2014). Retrieved from <https://knowledgetree.freshdesk.com/support/solutions/articles/5000011146-knowledgetree-backup-and-disaster-recovery-information>
53. Larman, C. (2003). *UML y Patrones. Introducción al análisis y diseño orientado a objeto*. (2da ed.). México : Prentice Hall, 2003.
54. Letelier, P., & Penadés, C. (2006). *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*, 05(1666-1680). Retrieved from <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
55. luckyBackup - backup and sync utility. (n.d.). Retrieved from <https://mislinuxapps.wordpress.com/?s=luckybackup>

56. luckyBackup - features. (n.d.). Retrieved from <http://luckybackup.sourceforge.net/manual.html>
57. LuckyBackup, tus backups nunca fueron tan simples. (n.d.). Retrieved from <http://ubunlog.com/luckybackup-tus-backups-nunca-fueron-tan-simples/>
58. Maira Mena Mugica. (2005). Gestión documental y organización de archivos. La Habana.
59. Marín, C. (2014). Archivólogo - blog de archivo - Lic. Retrieved from <http://archivologo.blogspot.com/2014/07/origen-de-la-gestion-documental.html>
60. MuyLinux. 21 herramientas de backup para Linux. (2009). Retrieved from <http://www.muylinux.com/2009/01/15/21-herramientas-de-backup-para-linux>
61. NetBeans. Writing JUnit Tests in NetBeans IDE. (n.d.). Retrieved from <https://netbeans.org/kb/docs/java/junit-intro.html>
62. OpenKm. OpenKm. OpenKm. (n.d.). Retrieved from <http://www.openkm.com>
63. Osmosis Latina. Respaldos ("Backups and Redundancy"). (n.d.). Retrieved from <http://www.osmosislatina.com/soporte/respaldos.htm>
64. Paper, W. (2014). Alfresco Backup and Disaster Recovery.
65. Papers, D. M. W. (n.d.). eclipse Cloud Document Management Software Features - docSTAR. Retrieved from <http://www.docstar.com/document-management-systems/eclipse/features>
66. Política de Backup de Seguridad. (n.d.).
67. Polo, M. (2010). Escuela Superior de Informática. Retrieved from <http://www.inf-cr.uclm.es/www/mpolo/tutorial/proposito.html>.
68. Portal Programas. Knowledgetree. (n.d.). Retrieved from <http://www.portalprogramas.com/knowledgetree/>
69. Pressman, R. S. (n.d.). Ingeniería del software. Un enfoque práctico. (4 ta).
70. Pressman, R. S., & Ph, D. (2010). Ingeniería del software. Un enfoque práctico. (7ma ed.).
71. Rafael Cid Ançay, Irina Álvarez Rodríguez, Alejandra Santiago Veiga, Nicolás, J. M. S. G. (n.d.). Comparación y Configuración de Backup Servers Systems.
72. Recovery Labs. Principales factores que causan una pérdida de información. (n.d.). Retrieved from <http://www.recoverylabs.com/ayuda-y-soporte/data-recovery-white-papers/informes/principales-factores-que-causan-una-perdida-de-informacion/>
73. Reviews of SmartFile Free Pricing & Demos Document Management Software. (n.d.). Retrieved from <http://www.capterra.com/document-management-software/spotlight/134146/SmartFile/SmartFile>
74. Rsync. (n.d.). Retrieved from <http://everythinglinux.org/rsync/>
75. Rsync. (n.d.). Retrieved from <http://samba.anu.edu.au/rsync/documentation>
76. Salas, C. (n.d.). Expertos en Sisitemas. Sistemas de Información y nuevas Tecnologías. Retrieved from <http://www.expertosensistemas.com/estrategias-de-copias-de-seguridad-iii/>
77. Santamar, R., Jos, C. J., & Gallego, C. (n.d.). Copias de seguridad. Estudio y metodología.

78. SANTANA, Y. (2011). Excriba como solución para la Gestión Documental. Ciudad de La Habana.: Ediciones Futuro.
79. Search Sotorage. PROVEEDORES DE TECNOLOGÍA DE ALMACENAMIENTO 2010. (n.d.). Retrieved from <http://www.searchstorage.es/acerca-de-nosotros/>
80. Seguridad, Tipos de Copias de. slideshare. Tipos de Copias de Seguridad. (n.d.). Retrieved from <http://es.slideshare.net/DimoniCukarell/tipos-de-copias-de-seguridad>.
81. Shields, I. (2009). Preparación para el examen 102 de LPI; Tema 111 Tareas administrativas. Retrieved from <http://www.ibm.com/developerworks/ssa/linux/edu/lpic1111/section6.html>
82. SmartFile _ File Sharing Solutions for Business & Enterprise. (n.d.). Retrieved from <http://www.smartfile.com/blog/>
83. Software Design Tools for Agile Teams, with UML, BPMN and More. (n.d.). Retrieved from <http://www.visual-paradigm.com/>
84. Software Terms. Java Definition. (2012). Retrieved from <http://techterms.com/definition/java>
85. Statistics about Leading Causes of Data Loss. (n.d.). Retrieved from <http://www.protect-data.com/information/statistics>
86. Sublime Text ¿el editor de texto definitivo? (n.d.). Retrieved from <http://www.genbetadev.com/herramientas/sublime-text-2-el-editor-de-texto-definitivo>
87. Sublime Text 2 ¿el editor de texto definitivo. (n.d.).
88. Sublime Text editor de texto para desarrolladores. (n.d.).
89. Swedorski, Scoot. Handy backup the power of simplicity. (n.d.). Retrieved from <http://www.handybackup.es/>
90. Unit Tests. (n.d.). Retrieved from <http://www.extremeprogramming.org/rules/unittests.html>
91. Universidad Nacional de Colombia, Facultad de Ciencias Económicas, Unidad Informática y Comunicaciones. Guía de Backups. (n.d.).
92. UniVerso El Periódico de los Universitarios - No. (n.d.).
93. What is integrated development environment (IDE)? - Definition from WhatIs.com. (n.d.). Retrieved from <http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>
94. XP - Extreme Programing Ingenieria de Software. (n.d.). Retrieved from http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html

Glosario de Términos

GDA: Gestión de Documentos Administrativos.

IBM: International Business Machines Corporation (empresa dedicada a la fabricación y comercialización de hardware y software).

CPU: Central Processing Unit (Unidad Central de Procesamiento)

SSH: Secure Shell (en español Intérprete de órdenes seguro)

SCP: Secure Copy Protocol (en español Protocolo de Copia Segura de archivos)

FTP: File Transfer Protocol (en español Protocolo de Transferencia de archivos)

XP: Extreme Programming (Programación Extrema)

CASE: Computer Aided Software Engineering (Ingeniería de Software Asistida por Computadora)

UML: Unified Modeling Language (en español Lenguaje Unificado de Modelado)

CRC: Clase-Responsabilidad-Colaborador

Plugins: programas que gestionan contenido de Internet

Snippet: pequeña porción de código

SAN: red de área de almacenamiento

Anexos

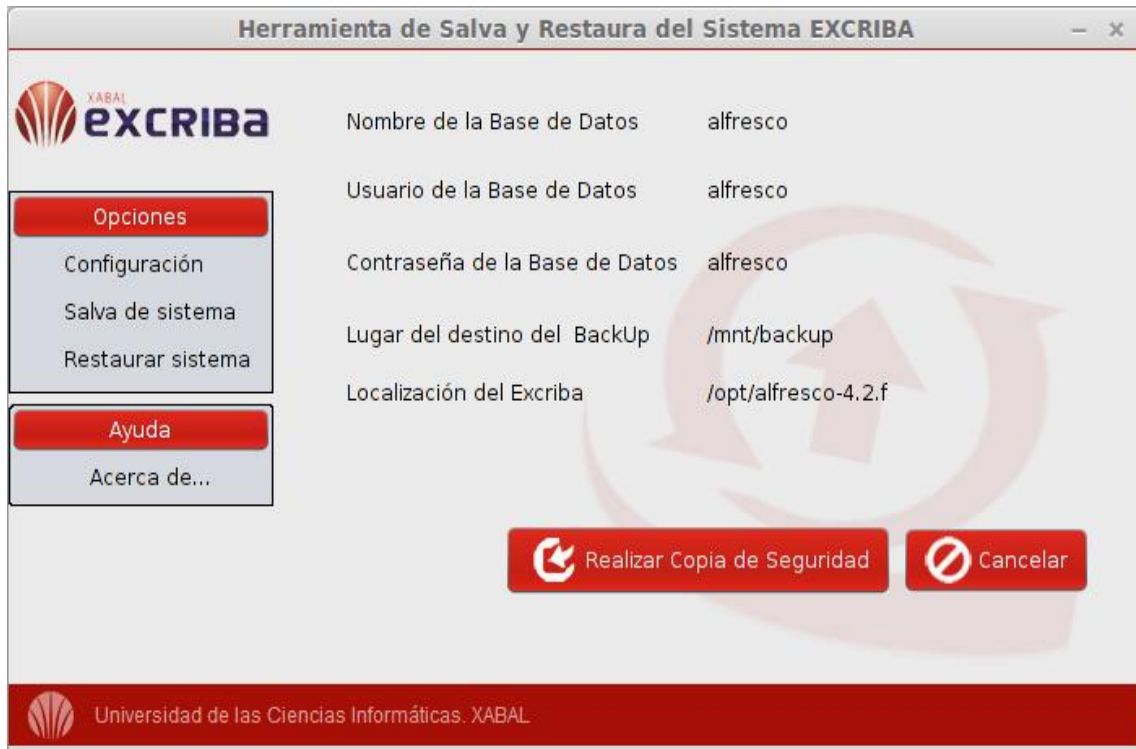


Figura 13 HU 5, 6

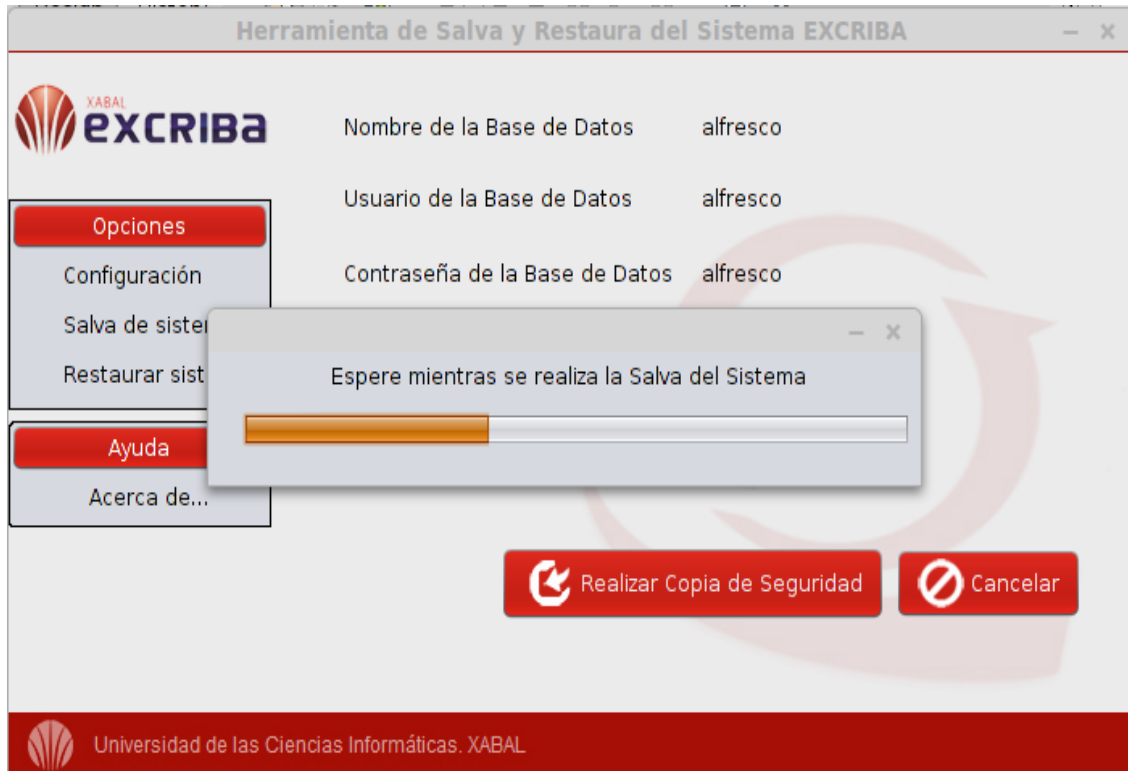


Figura 14 HU 6

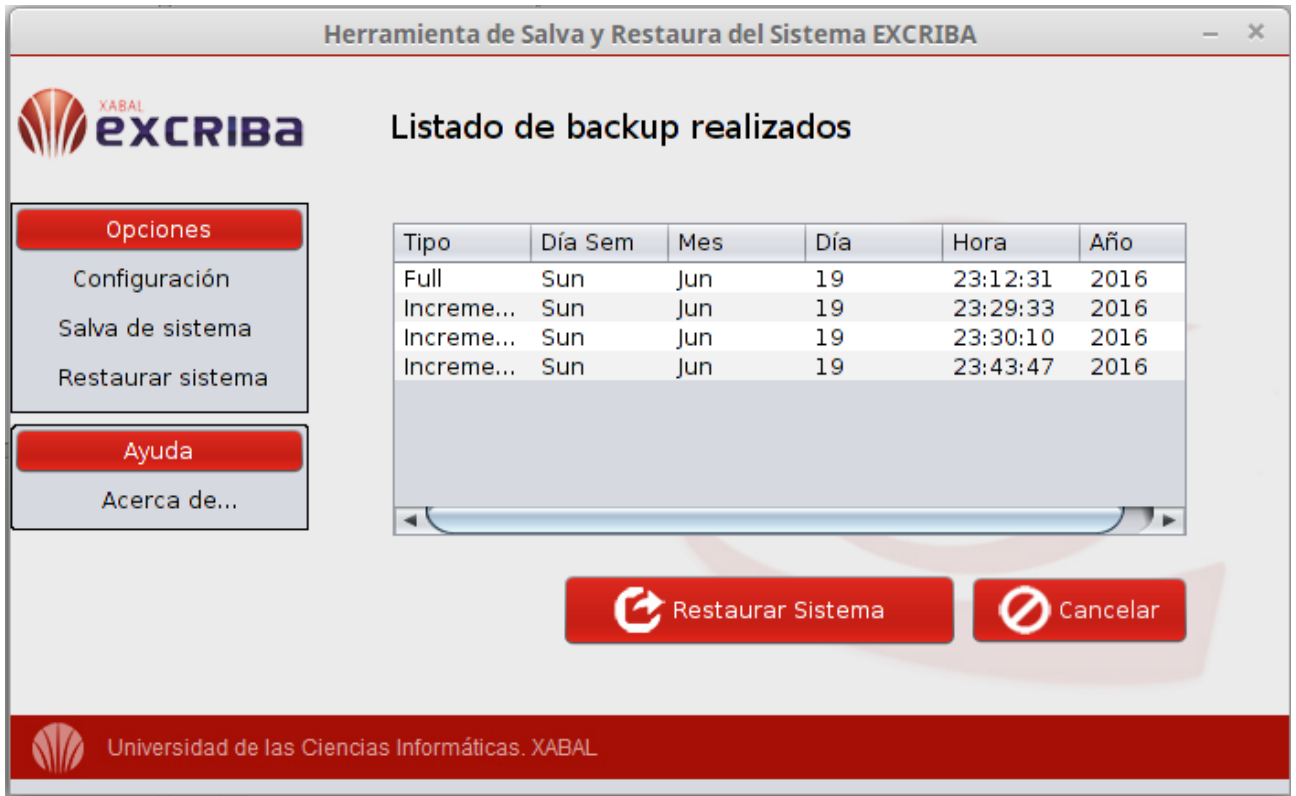


Figura 15 HU 7

```

public String Fecha(JTable tabla){
    String fecha=""; //A-M-D
    String Nmes="";
    String anno=(String) tabla.getValueAt(tabla.getSelectedRow(), 5);
    String mes=(String) tabla.getValueAt(tabla.getSelectedRow(), 2);
    String dia=(String) tabla.getValueAt(tabla.getSelectedRow(), 3);

    String[] meses={"Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"};

    for (int i = 0; i < meses.length; i++) {
        if (meses[i].equals(mes)&& i<10){
            Nmes="0"+String.valueOf(i+1);
            break;
        }
        else if (meses[i].equals(mes)&& i>=10){
            Nmes=String.valueOf(i+1);
            break;
        }
    }
}

```

Figura 16 Método fecha

```
private void jLabel3MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    String listabackup = prop.get_restore_list();
    String [] arg = listabackup.split("\n");
    String [][] mat = new String[arg.length][7];
    for (int i = 1; i < mat.length; i++) {
        if(!arg[i].contains("/")&&(arg[i]!="")){
            String [] f = arg[i].split(" ");
            for (int j = 0; j < f.length ; j++) {
                mat[i][j] = f[j];
            }
        }
    }

    String [] label = {"Tipo","Dia Sem","Mes","Dia","Hora","Año"};
    jTable2.setModel(new DefaultTableModel(mat, label));

    jPanel4.setVisible(false);
    jPanel3.setVisible(false);
    jPanel5.setVisible(true);
}
}
```

Figura 17 Método listar backup

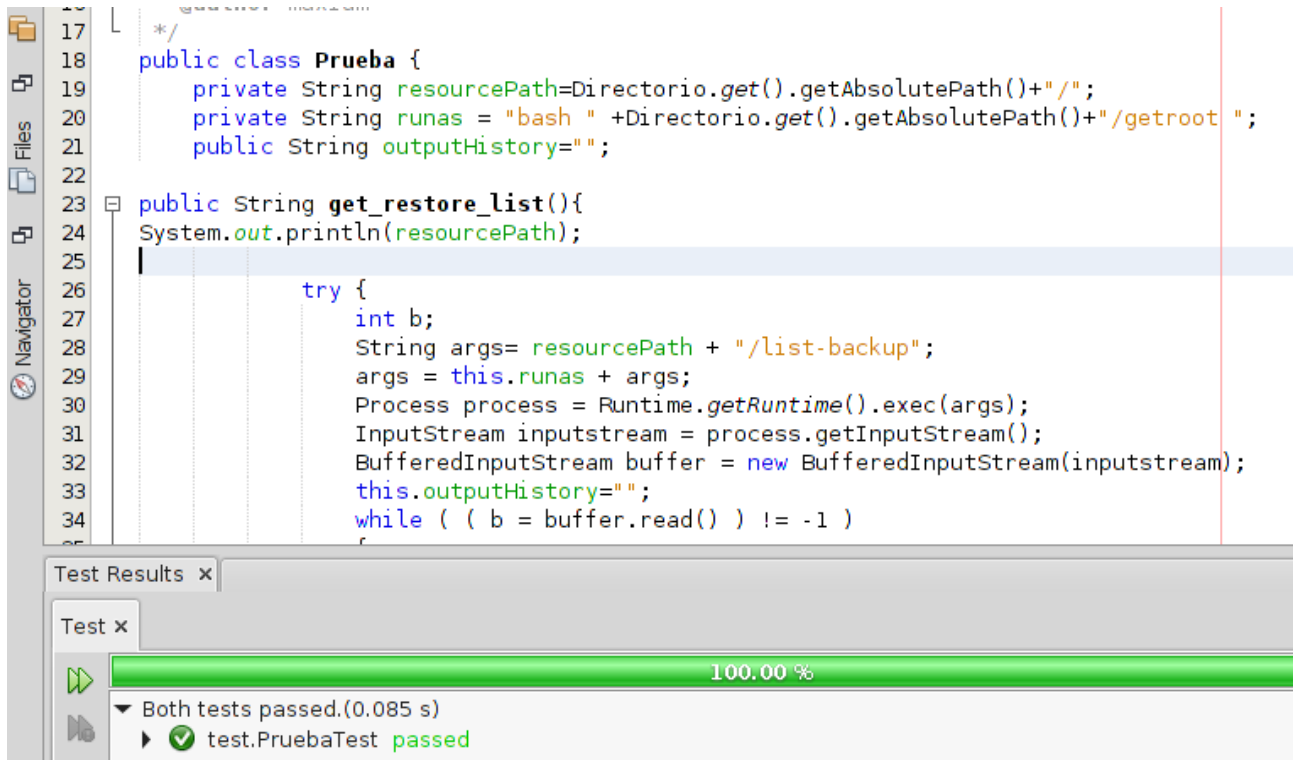


Figura 18 Resultado de la prueba unitaria al método `Get_restore_list` en JUnit.

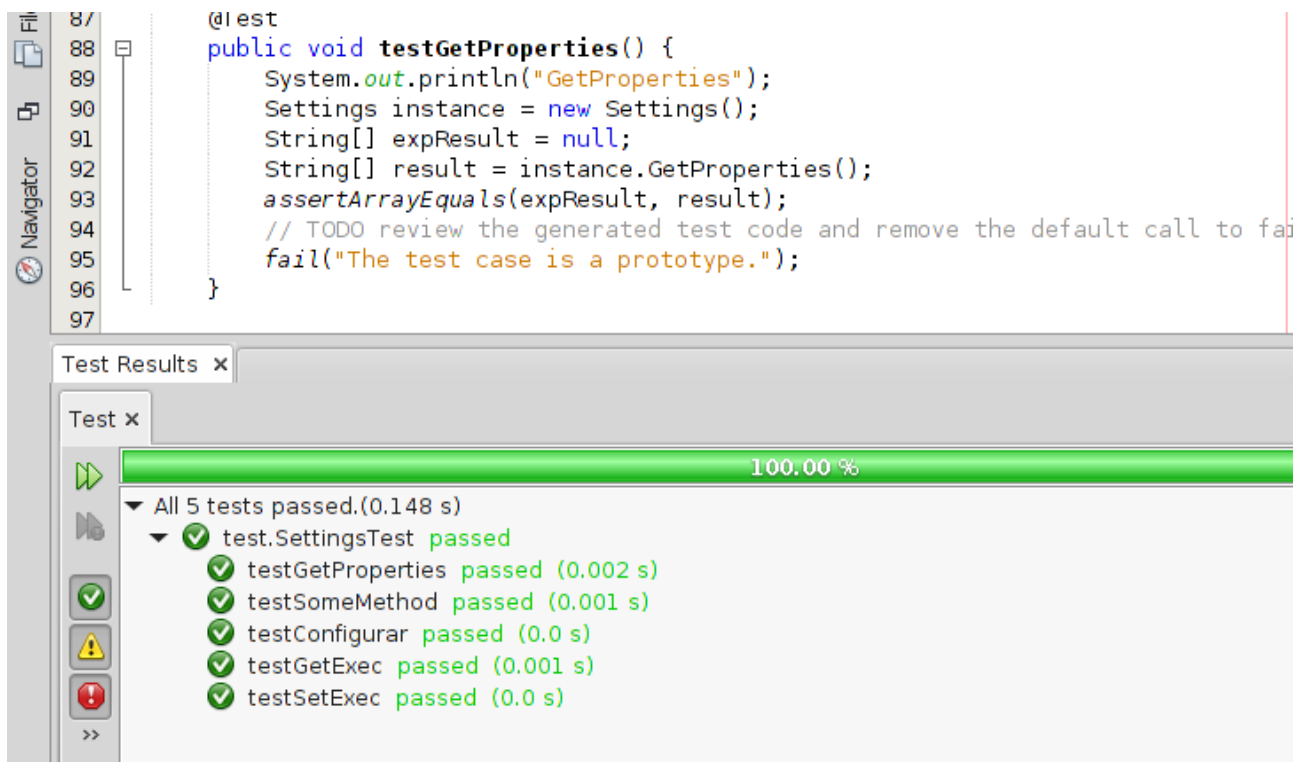


Figura 19 Resultado de la prueba unitaria al método `GetProperties` en JUnit.