

Universidad de las Ciencias Informáticas
FACULTAD 6



Título: Propuesta de métricas de Calidad para los procesos críticos del Grupo de Soporte en el Centro de Tecnologías de Gestión de Datos.

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

Autor: Luis Miguel Lobaina Cardero

Tutor: Ing. Niurka Martínez Durán

“La Habana, junio de 2012”

“Año 54 de la Revolución”



“Lo que no sea medible, hazlo medible”
Galileo Galilei

Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Luis Miguel Lobaina Cardero

Niurka Martínez Durán

Firma del Autor

Firma del Tutor

Datos de contacto

Autor: Luis Miguel Lobaina Cardero

E-mail: ldaliacq@infomed.sld.cu

Tutor: Niurka Martínez Durán

Ingeniero en Ciencias Informáticas

E-mail: nduran@uci.cu

Agradecimientos

Deseo agradecer a toda mi familia de manera general por todo su apoyo, pero existen personas que siempre han estado a mi lado y han jugado un papel importante en mi educación. Primeramente deseo agradecer a mi abuelo Miguel Eduardo, por toda su integridad y por inculcarme valores que han servido para formarme como hombre, si existe alguna recompensa para las personas buenas después de su muerte, estoy seguro que él está bien recompensado. Agradezco a mi abuela Nana por ser la mejor abuela del mundo y brindarme todo su cariño. Mi tía Yoly es la mejor tía, estoy seguro que todas las personas en esta vida desearían tener una tía tan especial como ella que siempre me ha apoyado y ha sido mi segunda madre.

Deseo agradecer a mi madre hermosa e inteligente, tanto ella como mi papá René se han esforzado en mi educación, la mayoría de mis logros son gracias a ellos, con toda sinceridad agradezco las cosas buenas que me inculcaron y me impusieron por mi bien. Mi madre ha sido mi mejor ejemplo académico y soy muy feliz de sentir que en ocasiones poseo parte de su inteligencia. A mi papá René le debo casi todo en la vida y aprovecho estas palabras para decirle que sus regaños nunca fueron en vano.

Agradezco a mi padre Guillermo y a su esposa Yanés por ser tan atentos conmigo y por apoyarme especialmente al principio de mi carrera.

Desde que tengo uso de razón he estado viendo a mi hermana Ida, ella siempre ha estado a mi lado mostrándome todo su apoyo junto a su esposo Rubén que se ha convertido en un hermano para mí. Para los dos, mis más sinceras muestras de gratitud.

Agradezco de todo corazón a mi prima Haydee por su apoyo incondicional especialmente durante toda mi carrera. Igualmente agradezco a mi primo Guille.

Agradezco a Anne por su amor y apoyo incondicional.

Agradezco a Ingrid por su dedicación y apoyo durante los años más difíciles de la carrera, gracias por enseñarme tantas cosas buenas.

Agradezco a Lily, por darme la oportunidad de poder contar con ella en las buenas y en las malas.

Agradezco a Yosmany por su amistad y por sus recomendaciones con el trabajo de diploma.

Agradezco a Yasmany por sus recomendaciones con Bonita Open Solution.

Agradezco a mis compañeras de estudio de segundo año Claudia y Juliet por su complicidad, su amistad y afinidad. Ellas son un ejemplo del buen estudio mezclado a la picardía e inteligencia.

Agradezco a mi tutora Niurka Martínez Durán por todo su apoyo y sus recomendaciones durante todo el proceso de tesis.

Agradezco a Dayris por su apoyo desde 4to año hasta la actualidad.

Dedicatoria

A mi familia por todo su apoyo incondicional, a mi madre en especial por ser mi mejor ejemplo académico.

Resumen

En la actualidad cada vez se hace más evidente la aplicación de métricas en proyectos y procesos informáticos. Las métricas brindan una clara visión del proceso en cualquiera de sus fases y permiten hacer estimaciones que conlleven a una mejora a largo plazo. En el Centro de Tecnologías de Gestión de Datos (DATEC) el Grupo de Soporte realiza una serie de procesos, que por su importancia contribuyen al buen funcionamiento de esta organización. Sin embargo, no existen elementos que proporcionen su medición. El presente trabajo de diploma tiene como objetivo desarrollar una propuesta de métricas de calidad para los procesos críticos de soporte en el centro DATEC que contribuya a una mejora continua, organización y correcta definición de los mismos, estos procesos se componen por la *Monitorización de Servidores*, las *Descargas de aplicaciones, herramientas y bibliografías* y la *Protección del código fuente*.

En el desarrollo de la investigación se realizó el modelado de los procesos utilizando la herramienta Visual Paradigm en su versión 8.0. Se diseñaron las métricas y para llevar a cabo su aplicación se simularon los procesos utilizando la herramienta Bonita Open Solution en su versión 5.6.2. Se realizaron mediciones en entorno real y se estableció una interpretación de los resultados obtenidos. La elaboración de una propuesta de métricas de calidad para los procesos críticos de soporte del centro DATEC permitió una mejor organización y correcta definición de estos procesos.

PALABRAS CLAVES:

Grupo de Soporte, mediciones, métricas, procesos críticos.

Índice de Contenido

Introducción.....	1
Capítulo 1: Fundamentos Teóricos.....	4
Introducción	4
1.1 Métricas de calidad	4
1.2. Métricas de calidad en los procesos.....	5
1.3.Estándares de métricas de calidad	6
1.3.1. ESTÁNDAR IEEE/EIA 12207	6
1.3.2. ESTÁNDAR ISO/IEC 9126 – MODELOS DE CALIDAD PARA SOFTWARE.....	7
1.3.3. ESTÁNDAR ISO/IEC 15939 MODELOS DE MEDICIÓN DE SOFTWARE	8
1.3.4. Estándar internacional (norma ISO 14764)	9
1.4 Herramienta para la modelación de procesos.....	11
1.5 Herramientas para la simulación de procesos	12
Conclusiones del capítulo.....	14
Capítulo 2: Diseño de métricas	15
Introducción	15
2.1. Procesos críticos del grupo de soporte.....	15
Definición de Proceso	15
2.1.1. Proceso de Monitorización de Servidores.....	15
2.1.2. Proceso de Descarga	21
2.1.3. Proceso de Protección del código fuente.....	26
2.1.4. Subproceso de Acceso al código	29
2.2. Establecimiento de métricas para los procesos críticos	31
2.2.1. Métricas en el proceso de Monitorización de Servidores	31
2.2.2. Métricas en el proceso de Descarga.....	33
2.2.3. Métricas en el subproceso de Gestión de las Listas de Control de Acceso (ACL). Proceso de Protección del código fuente.	37

2.2.4. Métricas en el subproceso de Acceso al código. Proceso de Protección del código fuente	40
2.3. Métricas generales	43
Conclusiones del capítulo.....	44
Capítulo 3. Aplicación de métricas	45
Introducción	45
3.1. Simulación.....	45
Imágenes del proceso de simulación	45
3.1.1. Simulación del proceso de “Monitorización de Servidores”	48
3.1.2. Comportamiento de los indicadores durante la simulación del proceso de Monitorización de servidores.....	49
3.1.3. Simulación del proceso de Descarga	50
3.1.4. Comportamiento de los indicadores durante la simulación del proceso de Descarga	50
3.1.5. Simulación del subproceso de Gestión de las Listas de Control de Acceso (ACL). Proceso de protección del código fuente.....	52
3.1.6. Comportamiento de los indicadores durante la simulación del subproceso de Gestión de las ACL.	52
3.1.7. Simulación del subproceso de Acceso al código. Proceso de protección del código fuente.	53
3.1.8. Comportamiento de los indicadores durante la simulación del subproceso de Acceso al código.	54
3.2. Proceso de mediciones en entorno real.....	55
3.2.1. Mediciones en entorno real para el proceso de Monitorización de Servidores	55
3.2.2. Mediciones en entorno real para el proceso de Descarga.....	56
3.2.3. Mediciones en entorno real para el proceso de Protección del código fuente. Subproceso de Gestión de las Listas de Control de Acceso (ACL)	57
3.2.4. Mediciones en entorno real para el proceso de Protección del código fuente. Subproceso de Acceso al código.....	59
3.3. Interpretaciones generales del proceso de mediciones en entorno real.....	60
Conclusiones del capítulo.....	63
Conclusiones Generales	64
Recomendaciones.....	65

Referencias Bibliográficas66
Bibliografía.....67

Índice de Figuras

Ilustración 1. Proceso de Monitorización de Servidores.....16

Ilustración 2. Subproceso Realizar Soporte a Servidores. Proceso de Monitorización de servidores.....17

Ilustración 3. Proceso de Descarga22

Ilustración 4. Subproceso Realizar descarga. Proceso de Descarga23

Ilustración 5. Subproceso Gestión de las Listas de Control de Acceso (ACL). Proceso de Protección del código fuente27

Ilustración 6. Subproceso Acceso al código. Proceso de Protección del código fuente29

Ilustración 7. Métrica: Espacio disponible en disco duro. Proceso de Monitorización de Servidores.....46

Ilustración 8. Métrica: Rapidez de la Descarga. Proceso de Descarga.....47

Ilustración 9. Métrica: Cantidad de permisos de acuerdo a Roles. Proceso de Protección del código fuente.....48

Índice de Tablas

Tabla 1 Descripción del proceso de Monitorización de Servidores	18
Tabla 2. Descripción del proceso de Descarga	23
Tabla 3. Descripción del subproceso de Gestión de las Listas de Control de Acceso (ACL)	27
Tabla 4. Descripción del subproceso de Acceso al código.....	30
Tabla 5. Métricas generales.....	43

Índice de Gráficos

Gráfico 1. Comportamiento de la métrica: Espacio disponible en disco duro. Proceso de Monitorización de Servidores.61

Gráfico 2. Comportamiento de la métrica: Rapidez de la Descarga. Proceso de Descarga62

Gráfico 3. Comportamiento de la métrica: Cantidad de permisos de acuerdo a roles. Proceso de Protección del código fuente.....62

Introducción

Con los avances de la tecnología el hombre se ha hecho dependiente de las computadoras y de los servicios asociados a estas. La actividad informática es cada vez más evidente en la vida cotidiana propiciando mejoras en la realización de las tareas. Es por ello que se debe garantizar en cada una de las organizaciones el buen funcionamiento de los equipos informáticos y de los servicios que se prestan a través de los mismos. Generalmente en la mayoría de las instituciones existe un Grupo de Soporte encargado de realizar tareas que contribuyan a un adecuado mantenimiento tecnológico.

Cada Grupo de Soporte Técnico debe brindar servicios, tales como: ayuda de una forma directa o indirecta al hardware o software de una computadora u otro dispositivo electrónico, este debe de estar disponible para prestar asistencia al usuario ante cualquier situación. Estos servicios se desglosan en procesos de soporte, que según su importancia, impacto o complejidad se convertirán o no en procesos críticos para cualquier entidad.

En este escenario Cuba presenta un adecuado nivel de actualización para garantizar el mantenimiento tecnológico en las diversas instituciones. La Universidad de las Ciencias Informáticas (UCI) es la organización de desarrollo de software que presenta la mayor infraestructura informática del país, se divide en diversos centros de desarrollo con sus propios Grupos de Soporte Técnico que responden a necesidades específicas.

El Centro de Tecnologías de Gestión de Datos (DATEC) forma parte de la infraestructura productiva de la UCI, este tiene como misión principal desarrollar soluciones integrales, productos y servicios relacionados con las tecnologías de gestión de datos y bioinformática; la formación de profesionales integrales con un alto nivel científico-técnico y productivo; y la realización de investigaciones afines.

DATEC presenta su propio Grupo de Soporte encargado de realizar procesos tales como: la gestión para reparar las averías que existan en los laboratorios, oficinas u otros locales del centro, también se gestiona el Movimiento de Medios tanto interno como externo, se Monitorizan los Servidores, se gestiona el proceso de Descarga de aplicaciones, herramientas y bibliografías, así como el Control y Gestión de Salvas, se redistribuye la Asignación de los Recursos de acuerdo a las necesidades de cada proyecto. Dentro de los procesos antes mencionados existen algunos que presentan mayor importancia con respecto a los otros, entre ellos figuran la Monitorización de Servidores, las Descargas de aplicaciones, herramientas y bibliografías y la Protección del código fuente; estos procesos se definen como críticos.

Cada uno de los procesos antes mencionados demanda una serie de actividades importantes debido a que garantizan la organización y contribuyen al buen funcionamiento del centro DATEC, haciendo énfasis en la esfera productiva. Sin embargo, en ocasiones los procesos se tornan incomprensibles e inestimables para fases posteriores y puede volverse engorroso establecer una valoración técnica de estos, fundamentalmente durante su transcurso.

La inexistencia de estadísticas y datos históricos en cuanto al desempeño de los procesos de soporte del centro DATEC ha impedido que se puedan establecer comparaciones, estimaciones e incluso tomar decisiones inmediatas para mejorar las salidas de estos. En ocasiones es necesario predecir nuevos estados de comportamiento y se carece de elementos que permitan ejercer estas acciones. Tampoco existe organización y documentación de las actividades que se llevan a cabo, trayendo consigo descontrol técnico.

Por la situación anteriormente descrita surge el siguiente **problema de la investigación**: ¿Cómo contribuir a la mejora de los procesos de soporte del centro DATEC?

Para brindarle solución a este problema se define como **objeto de estudio**: Propuesta de métricas de calidad y enmarcando como **campo de acción**: Métricas de calidad para los procesos de soporte.

Por tal motivo el **objetivo general** de esta investigación es desarrollar una propuesta de métricas de calidad para los procesos críticos de soporte en el Centro de Tecnologías de Gestión de Datos (DATEC) que contribuya a una mejora continua, organización y correcta definición de estos procesos.

Para dar cumplimiento al objetivo general se trazaron los siguientes **objetivos específicos**:

- Analizar estándares de métricas de calidad atendiendo al estado del arte y las condiciones propias del Centro de Tecnologías de Gestión de Datos (DATEC).
- Diseñar métricas de calidad para los procesos críticos de soporte del Centro de Tecnologías de Gestión de Datos.
- Aplicar las métricas obtenidas en el centro DATEC.

Para dar cumplimiento a los objetivos planteados se trazaron las siguientes **tareas de la investigación**:

- Análisis de las tendencias actuales sobre métricas para procesos de soporte.
- Análisis de estándares de calidad propuestos para el desarrollo de métricas.

- Descripción de los procesos Monitorización de servidores, Protección del código fuente y Descarga de herramientas, aplicaciones y bibliografías.
- Diseño de métricas de calidad para el proceso de Monitorización de servidores.
- Diseño de métricas de calidad para el proceso de Descarga.
- Diseño de métricas de calidad para el proceso de Protección del código fuente.
- Simulación de las métricas en los procesos críticos de soporte del centro DATEC.
- Mediciones en un entorno real de las métricas propuestas.

Como **posible resultado** se obtiene una propuesta de métricas de calidad para los procesos críticos de soporte en el Centro de Tecnología de Gestión de Datos (DATEC), permitiendo una organización y definición correcta de los procesos.

La presente investigación cuenta con introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y referencias bibliográficas.

Capítulo 1: Fundamentación Teórica.

Constituye la base teórica de la investigación realizada. Se realiza un análisis y se exponen conceptos relacionados al término de métricas, así como de los diferentes tipos existentes asociados al término de calidad específicamente para procesos informáticos. Se describen los diversos estándares internacionales y se selecciona uno de estos al igual que las herramientas para el desarrollo de las métricas.

Capítulo 2: Diseño de Métricas.

Se realiza la descripción de los procesos críticos de soporte del Centro de Tecnologías de Gestión de Datos (DATEC). Se establecen métricas para cada uno de estos procesos de acuerdo a sus características.

Capítulo 3: Aplicación de métricas.

Se lleva a cabo la aplicación de las métricas elaboradas para cada uno de los procesos críticos de soporte del centro de desarrollo DATEC. Para ello se simulan cada uno de estos procesos y se realizan mediciones en entorno real.

Capítulo 1: Fundamentos Teóricos

Introducción

En el presente capítulo se realiza un análisis y se exponen conceptos relacionados al término de métricas, así como de los diferentes tipos existentes asociados al término de calidad y desarrolladas para los procesos informáticos. Se describen los diversos estándares internacionales y se selecciona uno de estos al igual que las herramientas para el desarrollo de métricas.

1.1 Métricas de calidad

El IEEE "*Standard Glossary of Software Engineering Terms*" define la palabra **métrica** como "una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado". (1) Son escalas de unidades sobre las cuales puede medirse un atributo cuantificable. Se debe recopilar y analizar datos basándose en mediciones reales teniendo en cuenta la calidad del producto o servicio, así como a las escalas de medición. Los atributos son características observables del producto o de un proceso determinado.

De acuerdo a la investigación se hace necesario realizar una revisión de las diversas definiciones de calidad:

- **Real Academia Española** "Propiedad o conjunto de propiedades inherentes a una cosa, que permiten apreciarla como igual, mejor o peor que las restantes de su especie". (2)
- **La norma UNE-EN ISO 9000:2000** define calidad como el grado en el que un conjunto de características inherentes cumple con los requisitos (3)
- **IEEE** "Grado con el cual el cliente o usuario percibe que el software satisface sus expectativas". (3)
- **La norma ISO 8402** define la calidad como: "Totalidad de características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades expresadas o implícitas". Estas necesidades especificadas, bien pueden estar en un contrato o se deben definir explícitamente. El logro de la calidad puede tener tres orígenes: calidad realizada, calidad programada y calidad necesaria. La primera es la que es capaz de obtener la persona que realiza el trabajo, la segunda es la que ha pretendido obtener y la tercera la que exige el cliente y que le gustaría recibir. La gestión de la calidad pretenderá que estas coincidan. (4)

Una vez analizados los distintos conceptos se puede definir la calidad como el grado de satisfacción y conformidad del cliente de acuerdo con los requisitos. En el ámbito informático tiene su punto de

partida en la creación de sistemas de información. Debido a la importancia que tiene esta, surge la necesidad de medición en la gestión y el aseguramiento de la calidad del software.

Las métricas de calidad son un conjunto de reglas generadas que brindan una clara visión del proceso de negocio en cualquiera de sus fases. Proveen información necesaria para la toma de decisiones técnicas, también se usan para poder medir en términos contables la calidad de los procesos y evitar errores comunes durante el transcurso de estos.

1.2. Métricas de calidad en los procesos

Establecer métricas para procesos implica llevar a cabo mediciones dentro de las actividades. Estas permiten tener una visión profunda del proceso posibilitando la toma de decisiones, ayudan además a analizar el trabajo desarrollado, conocer si se ha mejorado o no con respecto a proyectos anteriores, contribuyen a detectar áreas con problemas para poder remediarlos a tiempo y a realizar mejores estimaciones.

Las métricas de proceso deben reflejar:

- Qué tan bien el proceso de desarrollo está siendo llevado a cabo en términos de puntos de revisión y en objetivos de calidad en el proceso, siendo cumplidos en términos de calendario.
- Qué tan efectivo es el proceso de desarrollo, al reducir la probabilidad que se introduzcan fallas o que cualquier falla sea detectada.

En las métricas de proceso los niveles deben ser conocidos y utilizados para el control del proceso en busca de mejoras para que no sean utilizadas métricas fijas. Las métricas seleccionadas deben de ajustarse al proceso utilizado y si es posible, tener un impacto directo sobre la calidad.

Para mejorar un proceso se deben medir los atributos del mismo, desarrollar métricas de acuerdo a estos atributos y utilizarlas para proporcionar indicadores que conduzcan a la mejora del proceso. Es importante destacar que todo lo recolectado e interpretado para establecer métricas de calidad en los procesos en una determinada organización debe tener cierta utilidad. En caso de no tenerla implicaría un riesgo para la organización.

Las métricas deben ser enunciadas y utilizadas para administrar el proceso de negocio. Con su creación se pretende recopilar y actuar sobre las medidas cuantitativas de la calidad. Estas medidas deben ser utilizadas con los propósitos siguientes:

- Recopilar información y reportar valores que sirvan para realizar comparaciones en el futuro.
- Identificar el actual nivel de desempeño por cada actividad de proceso.
- Llevar a cabo acciones que tributen a mejorar desde el punto de vista técnico las actividades dentro del proceso.
- Establecer metas de mejoras específicas a los procesos.

1.3. Estándares de métricas de calidad

Los estándares de calidad son importantes de acuerdo al uso que se les pueda dar en cuanto a: mejoramiento del producto, protección al comprador, protección al negocio. Estos a su vez incrementan la disciplina laboral de los productores y contribuyen de manera eficiente en la introducción de la tecnología jugando además un rol vital en la transición tecnológica. Contribuyen en gran medida al mejoramiento del producto, pueden ser voluntarios, la organización que los adopta mejora sus productos o la percepción de estos en el mercado.

Estas normas benefician los procesos de negocio permitiendo desarrollar la producción con costos más apropiados. Proveen al comprador de cierta protección debido a que con muchos productos disponibles es posible tomar decisiones basadas en propaganda, folletos, experiencias anteriores con el vendedor o examinación directa. La creciente complejidad de productos tecnológicos causa inevitablemente la imposibilidad de examinar muchos aspectos que se mantienen ocultos hasta después de ser adquiridos.

Existen diferentes motivos por los cuales una organización adopta los estándares en vista a mejorar y evaluar su capacidad tomando en cuenta los siguientes aspectos:

- Calidad
- Satisfacción del cliente
- Productividad
- Madurez de los procesos
- Tecnología

1.3.1. ESTÁNDAR IEEE/EIA 12207

Esta norma establece las directrices sobre el aseguramiento de la calidad de software. Busca que el cliente se sienta seguro de que el software cumple con los requerimientos que el usuario define: requerimientos reglamentarios, técnicos y de la organización. Define además los objetivos de calidad,

objetivos del proceso de aseguramiento de la calidad y cuáles son los procesos de aseguramiento de la calidad.

Características esenciales

- Establece un marco común para el software a través de su ciclo de vida, desde la concepción hasta el retiro del mismo.
- Enfoca los procesos del software desde el punto de vista técnico del sistema y desde el punto de vista comercial de la empresa.
- Es considerado ampliamente como base para el comercio mundial de software.
- Su adopción es completa o en camino de serla en los países más desarrollados.

Este estándar está orientado al aseguramiento de la calidad de software y no se encuentra orientado al proceso de mantenimiento de este, evitando de esta manera alguna posibilidad de adaptación para la creación de métricas para los procesos críticos de soporte del centro DATEC. A pesar de que está bien enmarcado en todo el proceso de calidad no está orientado al Proceso de Soporte.

1.3.2. ESTÁNDAR ISO/IEC 9126 – MODELOS DE CALIDAD PARA SOFTWARE

Este estándar está pensado para los desarrolladores, adquirentes, personal que asegure la calidad y evaluadores independientes, responsables de especificar y evaluar la calidad del producto de software. Por tanto, puede servir para validar la completitud de una definición de requisitos, identificar requisitos de calidad de software, objetivos de diseño y pruebas, criterios de aseguramiento de la calidad, etc. La calidad del software puede evaluarse midiendo los atributos internos (medidas estáticas o productos intermedios) o atributos externos (comportamiento del código cuando se ejecuta).

Esta norma se orienta a dos áreas para la calidad del producto de software, calidad interna - externa, y calidad en uso del producto de software. (6)

- ✓ Modelo de Calidad Interna y Externa; este modelo determina características y sub características, que sirven para evaluar la calidad interna y la calidad externa de un producto de software como sigue: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenimiento, portabilidad.
- ✓ Modelo de Calidad en Uso; este modelo propone métricas de calidad en uso categorizadas en las cuatro características siguientes: efectividad, productividad, seguridad física, satisfacción. (5)

El presente estándar a pesar de estar orientado para los desarrolladores de software establece parámetros tales como la validación de requisitos y la medición de atributos. Se tienen en cuenta aspectos como el mantenimiento de software, el cual forma parte del soporte técnico y se proponen métricas de calidad como parte del ciclo de desarrollo. Es importante destacar que aunque se proponen métricas de calidad el estándar ISO/IEC 9126 no cumple con las características para ser el eje rector de la propuesta de métricas a desarrollar para los procesos críticos del centro DATEC debido a que no propone métricas para el proceso de soporte.

1.3.3. ESTÁNDAR ISO/IEC 15939 MODELOS DE MEDICIÓN DE SOFTWARE

Este modelo sirve para evaluar los procesos de medición de software y se subdivide en dos grupos:

- Proceso de medición del software
- Modelo de información de la medición

El proceso de medición del software está dirigido por las necesidades de información de la organización. Por cada necesidad de información este proceso produce un producto que intenta satisfacerlo. El modelo de información de la medición establece el vínculo entre medidas y necesidades. Las entidades de la medición incluyen procesos, productos, proyectos y recursos. El modelo describe cómo los atributos relevantes se cuantifican y se convierten en indicadores que proporcionan una base para la toma de decisiones. Este estándar define los términos medida base y medida derivada como tipos particulares de medida. También adapta algunos términos para la valoración del proceso, como: “medición del proceso” y “objetivo cuantitativo”.

El estándar ISO/IEC 15939 identifica las actividades y las tareas necesarias para definir, seleccionar, aplicar y mejorar con éxito la medición del software dentro de un proyecto o una estructura de medición organizacional. También provee definiciones para términos de medición comúnmente usados dentro de la industria del software. (5)

Este estándar evalúa los procesos de medición del software, expresa además que los atributos relevantes se cuantifican y se convierten en indicadores proporcionando una base para la toma de decisiones. No obstante está muy orientado a la medición de software y no a la del proceso de soporte, por lo que no es el indicado para regir la elaboración de una propuesta de métricas de calidad para los procesos críticos del centro DATEC.

1.3.4. Estándar internacional (norma ISO 14764)

Este estándar internacional aclara los requerimientos para el proceso de Mantenimiento del Software. El Mantenimiento del Software es un proceso primario en el ciclo de vida de un producto de software tal como se describe en ISO/IEC 12207.

Describe en gran detalle la gestión del Proceso de Mantenimiento, el cual aparece en el estándar ISO/IEC 12207, y además establece definiciones para los distintos tipos de mantenimiento. Proporciona una guía aplicable a la planificación, ejecución, control, mantenimiento, revisión y evaluación de forma cercana al proceso de mantenimiento.

Brinda la posibilidad de ejecutar y evaluar planes de mantenimiento de software de manera general y específica, así como adaptar el alcance y magnitud de los productos de software. Este estándar internacional proporciona una estructura, terminología precisa y procesos que permiten la aplicación consistente de la tecnología (herramientas, técnicas y métodos) para el mantenimiento de software. (7)

La primera aparición de las métricas en este estándar internacional se evidencia en el Proceso de Mantenimiento, el cual recibe como entradas la descomposición de trabajos, informes de rendimiento, peticiones de cambio y plan de gestión de alcance. Este a su vez utiliza un sistema para el control de cambios, un conjunto de métricas y una planificación adicional. Tiene como salidas cambios concretos y acciones correctivas que se tienen que ejecutar. El uso de las métricas en este estándar internacional juega un papel decisivo en diversos aspectos:

- **Control de cambio del alcance.**

Utilizando un sistema para el control de cambios, un conjunto de métricas y una planificación adicional se devuelven los cambios concretos y acciones correctivas que se deben ejecutar (además de lecciones aprendidas para el futuro).

- **Medida del software. Planificación de la calidad.**

Aplicando las técnicas del análisis de costo/beneficio, comprobación del rendimiento usando métricas, diseño de experimentos y construcción de diagramas de flujo a la política de calidad y a la descripción del producto se obtendrá un plan para la gestión de la calidad y unas listas de comprobación.

- **Medida del software. Aseguramiento de calidad.**

Se conseguirán mejoras al utilizar técnicas de auditoría para el plan de gestión de calidad sobre la base de los resultados de las métricas.

- **Documentación del proceso.**

El proceso de Mantenimiento detallado debe documentarse de forma que todo el personal siga la misma secuencia. Las métricas deben apoyar el proceso y los esfuerzos para la mejora del software.

- **Mantenibilidad y el proceso de desarrollo.**

Los requerimientos cuantitativos se usan para definir las magnitudes de mantenibilidad o criterios de calidad así como las métricas usadas para determinar valores claves durante las distintas fases del ciclo de vida.

- **Designación de la persona encargada del mantenimiento. Planificación.**

Existen métricas para la mantenibilidad de esfuerzo (evalúa el esfuerzo requerido durante la fase de mantenimiento), de complejidad y de estructura (analiza la correlación entre la estructura de un programa y su facilidad de mantenimiento).

- **Estimación de los costes de mantenimiento.**

Las métricas históricas deberán usarse como entradas para estimar los costes de mantenimiento siendo un dato cuantitativo importante para establecer estadísticas y prevenir aspectos futuristas en beneficio de una aplicación eficaz de los costes a la hora de realizar el mantenimiento.

- **Guías para el plan de mantenimiento.**

A la hora de realizar registros e informes de mantenimiento, de las actividades de mantenimiento se derivan métricas que son usadas posteriormente para establecer comparaciones y optimizar estas actividades en el futuro.

El estándar internacional (norma ISO 14764) describe en gran detalle la gestión del Proceso de Mantenimiento descrito en ISO/IEC 12207, y además establece definiciones para los distintos tipos de mantenimiento, proporciona una guía aplicable a la planificación, ejecución y control, mantenimiento, revisión y evaluación y de forma cercana al proceso de mantenimiento. Este estándar internacional describe el esqueleto del Proceso de Mantenimiento de Software pero no especifica los detalles de cómo implementar o ejecutar las actividades y tareas incluidas en el proceso. De acuerdo al propósito de cada uno de los estándares anteriormente mencionados, la norma ISO 14764 es el estándar más acorde para llevar a cabo la confección de métricas de calidad

para los procesos críticos de soporte del centro DATEC, debido a que está más orientado a la gestión del proceso de mantenimiento guardando una mayor relación con el soporte técnico. Para llevar a cabo el diseño de las métricas se seleccionaron diversos aspectos del presente estándar, tales como:

- Documentación del proceso.
- Mantenibilidad y el proceso de desarrollo.
- Designación de la persona encargada del mantenimiento. Planificación.
- Guías para el plan de mantenimiento.

1.4 Herramienta para la modelación de procesos

Visual Paradigm

Visual Paradigm es una de las herramientas UML CASE del mercado, considerada muy completa y fácil de usar, con soporte multiplataforma y que proporciona excelentes facilidades de interoperabilidad con otras aplicaciones.

Fue creada para el ciclo vital completo del desarrollo de software que lo automatiza y acelera, permitiendo la captura de requisitos, análisis, diseño e implementación. Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de base de datos a partir del esquema de las clases.

Está diseñada para usuarios interesados en sistemas de software de gran escala con el uso del acercamiento orientado a objeto, además apoya los estándares más recientes de las notaciones de Java y de UML. Incorpora el soporte para trabajo en equipo, posibilitando que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros. Visual Paradigm permite la Notación y Modelado de Procesos de Negocios de forma fácil y casi intuitiva. (8)

Características

- Software libre.
- Disponibilidad en múltiples plataformas
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.

- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa e inversa.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Licencia: gratuita y comercial.
- Soporta aplicaciones Web.
- Varios idiomas.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Generador de informes.
- Integración con Visio.

1.5 Herramientas para la simulación de procesos

✓ **ADONIS (BOC ITC GMBH)**

ADONIS permite el Modelado de procesos según una variedad de objetos predefinida y personalizada, tanto a nivel gráfico como a nivel de datos (atributos). El componente de simulación de ADONIS proporciona una simulación de eventos discretos con una biblioteca de cuatro algoritmos de simulación. ADONIS soporta lenguajes de modelado estándares como BPMN, UML, EPC, y LOVEM. Además el módulo ADL (ADONIS Definition Language) proporciona una tecnología de modelado que permite a los usuarios definir nuevos lenguajes de modelado. (9)

✓ **ARIS Simulation (IDS Scheer)**

ARIS se apoya en los procesos modelados por la herramienta ARIS Business Architect. Para ello, el usuario tendrá previamente:

- Informe de los atributos de simulación que quiere seguir (costes, plazo de tratamiento, plazo de espera).

- Definidas las funciones del proceso y los recursos humanos y materiales.
- Definida la disponibilidad de estos recursos en volumen y en tiempo.
- Definida la cantidad y la repartición de los datos que hay que tratar sobre el período simulado.

ARIS Simulation se apoya en el motor de simulación E-M Plant. Ofrece un motor de eventos discretos que proporciona un número amplio de capacidades de simulación, incluyendo la capacidad de definir y de reproducir múltiples eventos de procesos. Soporta la técnica EPC para modelar los procesos, así como también BPMN y UML. (9)

✓ **Bonita Open Solution (BOS)**

Bonita Open Solution (BOS) es una herramienta para la gestión de procesos BPM. Ha sido creada por BonitaSoft, una de las empresas principales en este mercado. BOS es una herramienta de código abierto, editor y solución de referencia de su tipo, multiplataforma y compatible con el estándar BPMN. Posee ciertas facilidades propias de un entorno de desarrollo ya que está desarrollado sobre la reconocida plataforma Eclipse. Su código puede ser descargado bajo la licencia GPL v2. (10)

Características principales:

- Código abierto
- Es ligera
- Compatibilidad con BPMN 2.0
- Interfaz Intuitiva
- Potente
- Fácil importación de procesos desde otras herramientas
- Personalización de las herramientas
- Integración de los procesos con reglas de negocio (10)

Dentro del mundo del software libre BOS es el primer editor y solución BPM de referencia. La empresa BonitaSoft, creadora de la suite, fue constituida en el año 2009 por los fundadores del proyecto de código abierto en cuestión. Uno de los objetivos que se persigue con BonitaSoft es ofrecer una solución fácil e intuitiva para la gestión de los procesos de negocio.

Para las empresas, Bonita Open Solution implica un aumento de productividad de las mismas, pues los usuarios son informados exactamente con lo que deben hacer, esto permite una mayor eficacia en la ejecución de sus funciones en la medida que aprovechen la información brindada para organizarse. Al ser software libre permite una reducción considerable en aspectos como el tiempo, el esfuerzo y el presupuesto destinado a la ejecución del proyecto. Las empresas pueden beneficiarse con el hecho de que el gestor posibilita llevar a cabo las pruebas de diferentes configuraciones para la optimización de los procesos, obtener resultados en poco tiempo y medir los progresos realizados. Además las actividades pueden ser monitoreadas o seguidas en tiempo real para llegar a conclusiones sobre su eficiencia. (11)

Luego de analizar las distintas herramientas se escogió para realizar la simulación de los procesos a Bonita Open Solution en su versión 5.6.2 debido a que es código abierto, presenta además una interfaz intuitiva, manejable y fácil de usar. Es ligera y potente a la vez, siendo esto un aspecto favorable para su uso. Durante la simulación BOS permite establecer parámetros análogos a situaciones reales los cuales pueden ser instanciados tantas veces como desee el usuario. A pesar de que en la presente investigación no se pretende hacer uso de la herramienta en su plenitud sí se pretende llevar a cabo todo el modelado de los procesos críticos de soporte del centro DATEC y la simulación de estos a partir de las métricas que se establezcan durante el transcurso de la investigación.

Conclusiones del capítulo

En el presente capítulo se realizó un estudio de las métricas de calidad haciendo énfasis en los procesos. Se analizaron diversos estándares, seleccionándose la norma ISO 14764 como base para la creación de la propuesta de métricas para los procesos críticos del grupo de soporte del centro DATEC. Se seleccionaron como herramientas de modelado y simulación de procesos a Visual Paradigm en su versión 8.0 y a Bonita Open Solution en su versión 5.6.2 respectivamente.

Capítulo 2: Diseño de métricas

Introducción

En el presente capítulo se lleva a cabo una breve descripción del concepto de proceso. Se realiza además el modelado de los procesos críticos del grupo de soporte del centro DATEC con una descripción textual de estos. Se elaboran las métricas para cada uno de estos procesos a partir de lo establecido en **la norma ISO14764**.

2.1. Procesos críticos del grupo de soporte

Definición de Proceso

La noción de proceso halla su raíz en el término de origen latino *processus*. Según el diccionario de la *Real Academia Española (RAE)*, este concepto describe la acción de avanzar o ir para adelante, al paso del tiempo, al conjunto de etapas sucesivas advertidas en un fenómeno natural o necesario para concretar una operación artificial. (12)

Aterrizando el concepto antes mencionado al ámbito informático se puede decir que:

Un proceso es un conjunto de acciones y actividades interrelacionadas que se llevan a cabo para alcanzar un conjunto previamente especificado de productos, resultados o servicios. (13)

Entre los procesos críticos que se llevan a cabo en el grupo de soporte figuran la Monitorización de Servidores, las Descargas de aplicaciones, herramientas y bibliografías y la Protección del código fuente.

2.1.1. Proceso de Monitorización de Servidores

La Monitorización de Servidores en tiempo real garantiza la disponibilidad e integridad de la información, permite conocer las vulnerabilidades del sistema mejorando de esta manera su protección. Esta actividad provee al centro DATEC de un control constante sobre el estado de los servidores pertenecientes a cada una de las líneas productivas, permitiendo que se puedan tomar medidas ante cualquier indicio de amenaza.

El punto de partida de este proceso es la conexión con el servidor central desde una estación de trabajo localizada en el centro DATEC, para obtener una visión de los reportes actuales del estado de los servidores. Una vez visualizados los reportes, se decide si se lleva a cabo o no la realización del

subproceso Realizar Soporte a Servidores. Realizándose o no este subproceso concluye el proceso de Monitorización de Servidores.

En el subproceso Realizar Soporte a Servidores se llevan a cabo actividades como: verificar el estado de los servidores de acuerdo al espacio disponible en los discos duros, se verifican los estados de las memorias Memoria de Acceso Aleatorio (RAM) y Memoria de Intercambio (SWAP), se chequea el estado de la Unidad Central de Procesamiento (CPU), se revisa el consumo de espacio en memoria de los procesos del Sistema Operativo (SO), se verifica además la existencia de actualizaciones para el Sistema Operativo. Una de las actividades que también se llevan a cabo en este subproceso es la verificación de la existencia de procesos zombies, este tipo de proceso del Sistema Operativo se caracteriza por haber terminado su ejecución sin aún haber sido eliminado de la CPU.

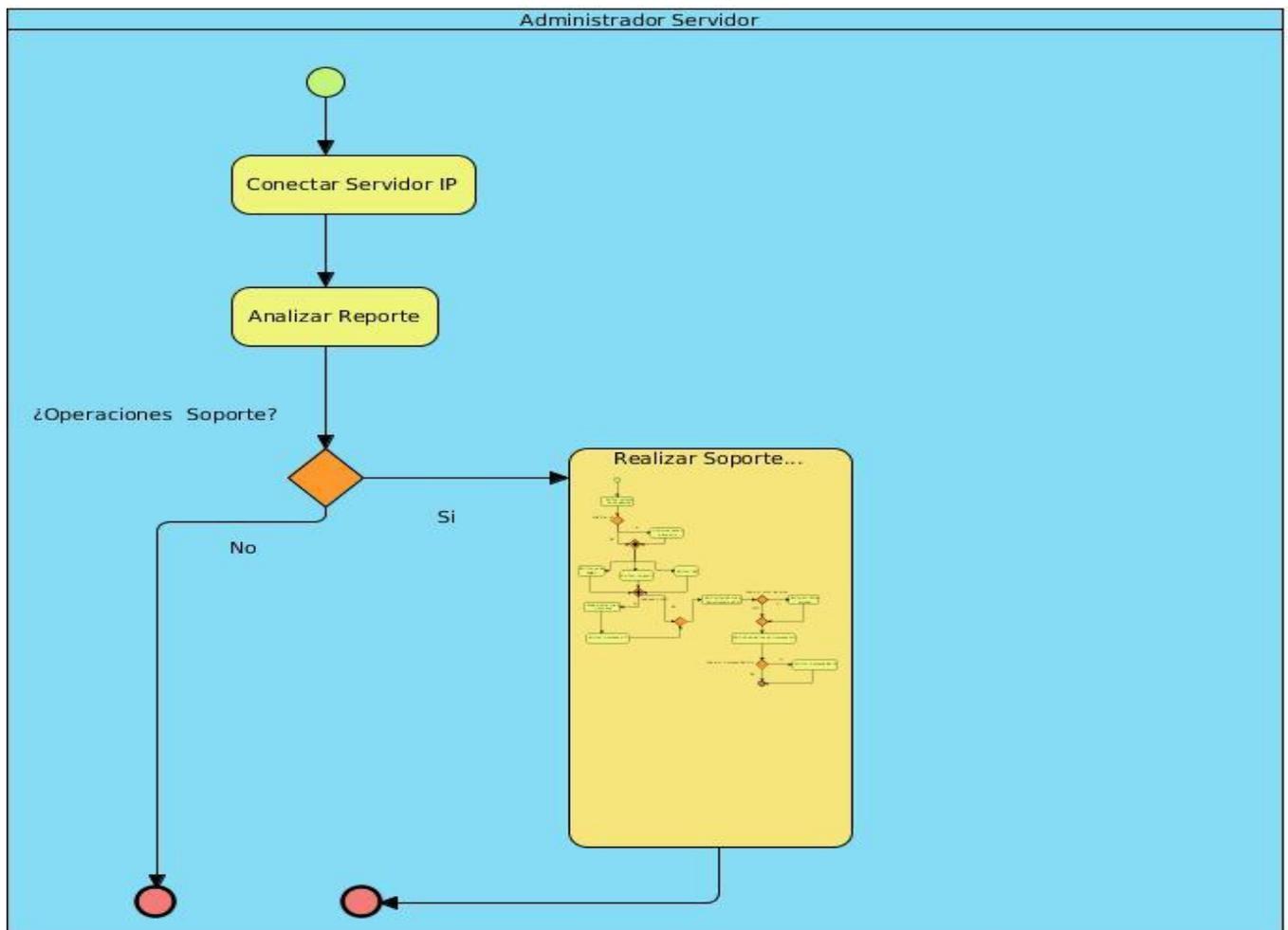


Ilustración 1. Proceso de Monitorización de Servidores

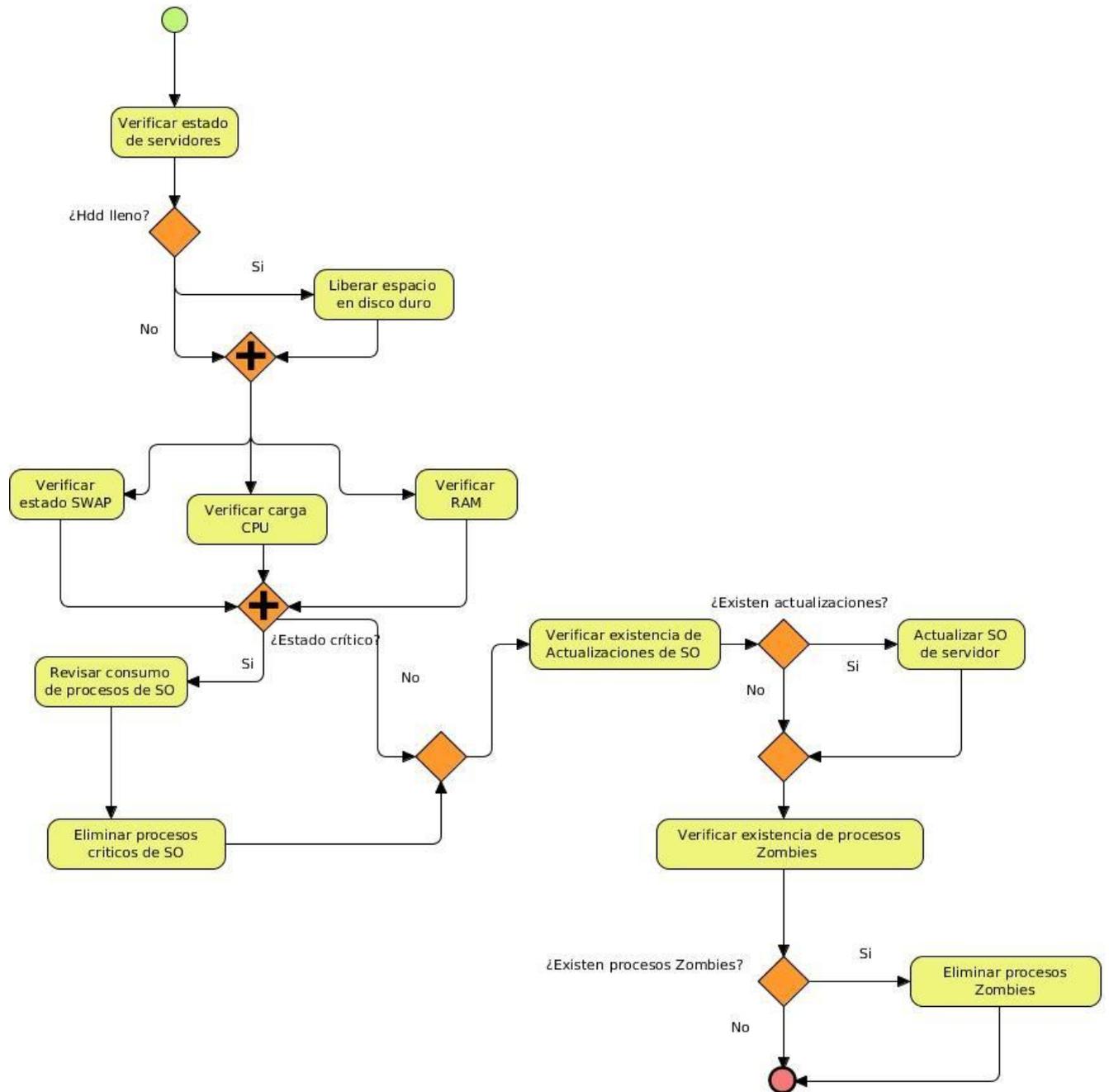


Ilustración 2. Subproceso Realizar Soporte a Servidores. Proceso de Monitorización de servidores

La tabla que se muestra a continuación brinda aspectos importantes del proceso de Monitorización de Servidores, tales como: su objetivo, persona responsable y persona que lo ejecuta. Se describe de forma enumerada el proceso teniendo en cuenta todos sus posibles estados. Se enuncian todas las actividades, mencionando para cada una de estas las tareas asociadas.

Tabla 1 Descripción del proceso de Monitorización de Servidores

<p>Proceso: Monitorización de Servidores</p>
<p>Objetivo: Mantener un adecuado funcionamiento de los servidores pertenecientes a cada una de las líneas productivas del centro DATEC.</p> <p>Responsable: Administrador de Servidores del Grupo de Soporte del Centro DATEC.</p>
<p>Ejecutor: Administrador de Servidores del Grupo de Soporte del Centro DATEC.</p>
<p>Descripción del proceso:</p> <p>1- El Administrador de Servidores se conecta desde su estación de trabajo al servidor que tiene asignado en la Infraestructura Productiva para la monitorización de servidores.</p> <p>2-Se analiza el reporte brindado a través de un servicio web.</p> <p>2.1-En caso de que el Administrador de Servidores esté de acuerdo con lo que se visualice en la monitorización decide no realizar operaciones y pasar a la actividad 3.</p> <p>2.2-En caso de que el Administrador de Servidores no esté de acuerdo con lo que se visualice en la monitorización decide pasar a la actividad 4.</p> <p>3- No se realizan operaciones manuales de soporte y concluye el proceso.</p> <p>4-Se procede a realizar operaciones de soporte dando inicio al subproceso "Realizar Soporte a Servidores", que recibe como entrada la verificación de los servidores de DATEC.</p> <p>4.1-En caso de que el disco duro del servidor esté lleno realizar la actividad 5.</p> <p>4.2-Si el disco duro aún tiene capacidad de almacenamiento se procede a pasar a la actividad 6,7 o la actividad 8 debido a que son actividades que pueden ocurrir en cualquier orden.</p> <p>5-Se procede a liberar espacio en el disco duro.</p> <p>6- Se verifica el estado de la memoria SWAP.</p>

- 6.1-**En caso de presentar un estado crítico realizar la actividad 9.
- 6.2-** En caso de presentar un estado normal realizar la actividad 11.
- 7-**Se verifica el estado del CPU.
- 7.1-**En caso de presentar un estado crítico realizar la actividad 9.
- 7.2-** En caso de presentar un estado normal realizar la actividad 11.
- 8-** Se verifica el estado de la memoria RAM.
- 8.1-**En caso de presentar un estado crítico realizar la actividad 9.
- 8.2-** En caso de presentar un estado normal realizar la actividad 11.
- 9-** Se procede a revisar el estado y consumo de espacio en memoria de los procesos del Sistema Operativo correspondientes al servidor.
- 10-**Se eliminan los procesos críticos del Sistema Operativo que consuman recursos innecesariamente.
- 11-**Se procede a verificar la existencia de actualizaciones en el Sistema Operativo.
- 11.1-** En caso de existir actualizaciones disponibles realizar la actividad 12.
- 11.2-** En caso de no existir actualizaciones disponibles realizar la actividad 13.
- 12-** Se actualiza el Sistema Operativo con las actualizaciones disponibles.
- 13-**Se verifica la existencia de procesos zombies.
- 13.1-**En caso de existir procesos zombies realizar la actividad 14.
- 13.2-**En caso de no existir procesos zombies se procede a finalizar el subproceso, con esta actividad concluye el proceso de Monitorización de Servidores.
- 14-**Se procede a eliminar los procesos zombies, con esta actividad concluye el subproceso Realizar Soporte a Servidores finalizando de esta manera el proceso de Monitorización de Servidores.

Proceso de Monitorización de Servidores		
No.	Actividades	Tareas
1	Conectar con servidor IP	1- Se realiza la conexión desde la estación de trabajo al servidor localizado en la Infraestructura productiva de la universidad.
2	Visualizar reporte	1- Se visualiza el reporte brindado mediante un servicio web. 2- Se analiza si es necesario realizar operaciones de soporte en los servidores. 3- En caso de no realizar operaciones de soporte el proceso concluye.
3	Subproceso: Realizar Operaciones de Soporte	1- Se procede a realizar el subproceso Realizar Operaciones de Soporte.
Subproceso Realizar Operaciones de Soporte. Proceso de Monitorización de Servidores.		
No.	Actividades	Tareas
1	Realizar operaciones de soporte	1- Esta actividad tiene como entrada la verificación del estado de los servidores. 2- En caso que el disco duro esté lleno se procede a liberar espacio en este.
2	Liberar espacio en disco duro	1- Se libera espacio en el disco duro
3	Verificar estado de la memoria SWAP	1- Se verifica la memoria en relación a los procesos que se ejecutan en el sistema.
4	Verificar estado del CPU	1- Se verifica el estado del CPU en relación a los procesos que se ejecutan en el sistema.
5	Verificar estado de la memoria RAM	1- Se verifica el estado de la memoria RAM en cuanto al espacio que han consumido los procesos en memoria.

6	Revisar consumo de los procesos	1- Se revisa cuáles son los procesos que están consumiendo mayor cantidad de recursos en cuanto a espacio en memoria.
7	Eliminar procesos críticos	1- Se eliminan los procesos innecesarios que estén consumiendo espacio en memoria.
8	Verificar existencia de actualizaciones en el SO	1- Se verifica para cuál de los servidores existen actualizaciones de seguridad disponibles para el sistema operativo.
9	Actualizar SO	1- Se actualiza el sistema operativo. 2- Se realiza el número de actualizaciones que se encuentren disponibles.
10	Verificar existencia de procesos zombies	1- Se verifica la existencia de procesos zombies. 2- Se verifica el número de procesos que se encuentren en ese estado.
11	Eliminar procesos zombies	1- Se eliminan los procesos zombies y concluye el subproceso.

2.1.2. Proceso de Descarga

Las Descargas de aplicaciones, herramientas y bibliografías permiten un desarrollo continuo en cuanto a la tecnología. De esta manera se pueden actualizar las herramientas usadas durante el desarrollo de los proyectos que se llevan a cabo en el centro DATEC. Este proceso posibilita obtener la bibliografía adecuada para el desarrollo de cada línea productiva, beneficiando en gran medida a cada proyecto del centro.

El proceso de Descarga tiene su punto de partida cuando un miembro de un determinado proyecto realiza una solicitud de descarga al Asesor de Soporte del centro DATEC a través de un correo electrónico, en el correo se especifica la dirección del archivo solicitado. El Asesor de Soporte procede a reenviar esta solicitud al Especialista de Seguridad Informática. Una vez recepcionada la petición el Especialista de Seguridad Informática procede a descargarla llevando a cabo el subproceso Realizar Descarga. El subproceso Realizar Descarga comienza cuando se recepciona la solicitud enviada por el Asesor de Soporte y finaliza cuando el archivo solicitado es descargado y luego almacenado en el servidor central de descargas de la universidad.

Una vez descargada la petición el Especialista de Seguridad Informática notifica al Asesor de Soporte el cual accede al servidor donde se encuentra el archivo solicitado y procede a copiarlo hacia el servidor de descargas del centro DATEC. El Asesor de Soporte notifica al Miembro del Proyecto por vía correo electrónico para que acceda a copiar el archivo. En caso de que el archivo solicitado presente algún problema el Miembro del Proyecto notifica al Asesor de Soporte y comienza el proceso nuevamente, en caso contrario finaliza el proceso de Descarga.

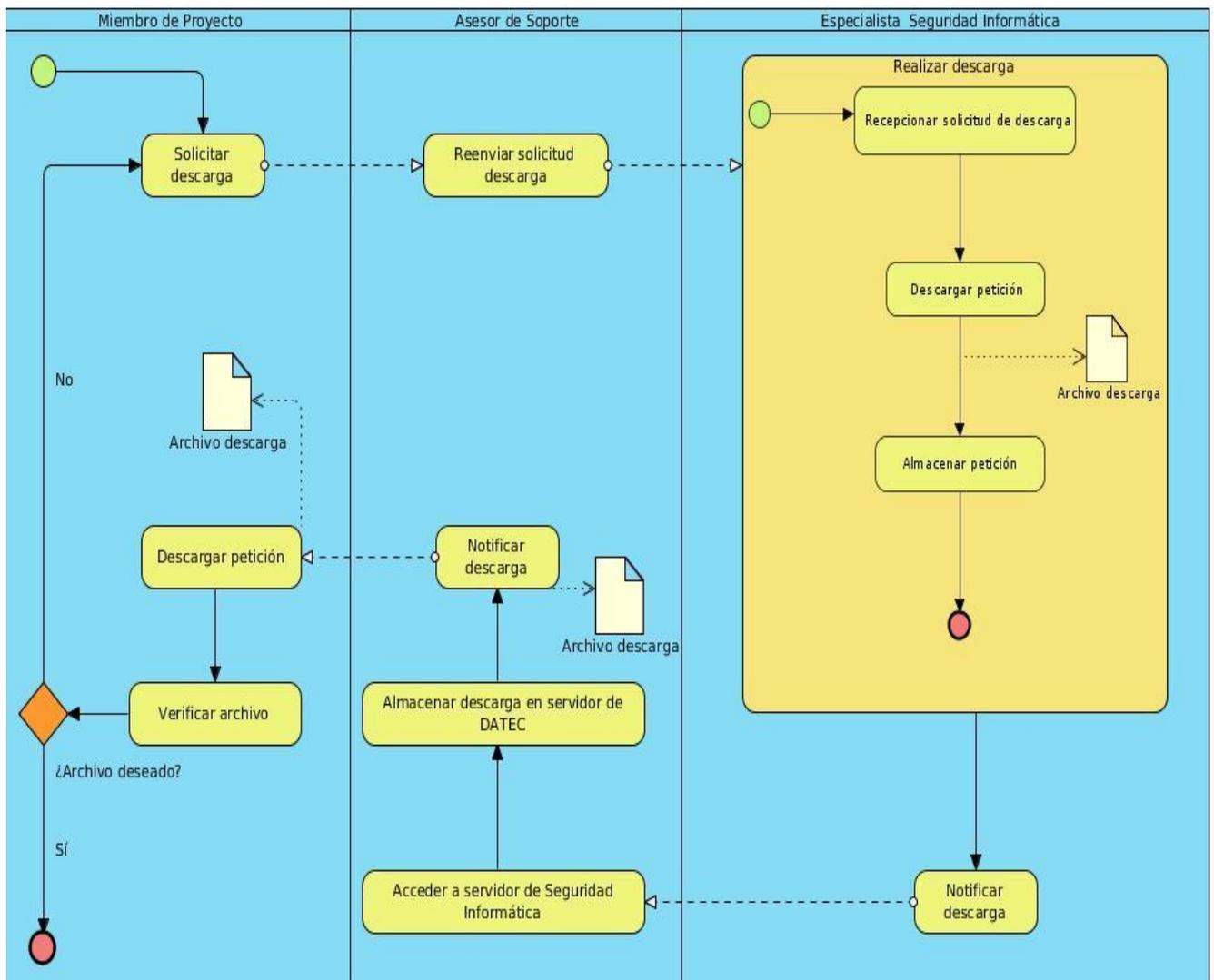


Ilustración 3. Proceso de Descarga

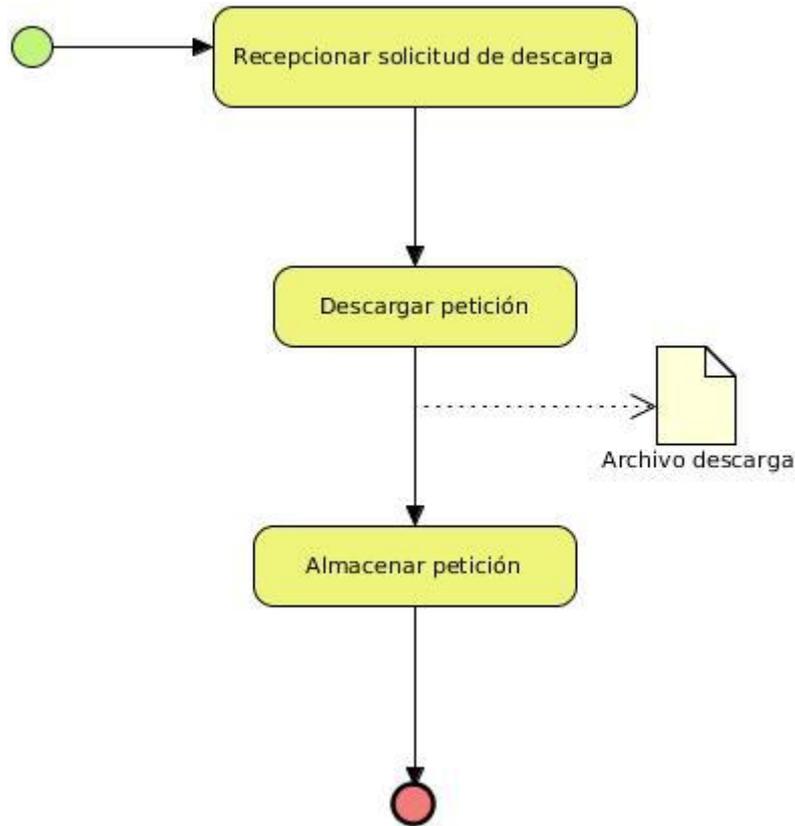


Ilustración 4. Subproceso Realizar descarga. Proceso de Descarga

La tabla que se muestra a continuación brinda aspectos importantes del proceso de Descarga, tales como: su objetivo, persona responsable y persona que lo ejecuta. Se describe de forma enumerada el proceso teniendo en cuenta todos sus posibles estados. Se enuncian todas las actividades, mencionando para cada una de estas las tareas asociadas.

Tabla 2. Descripción del proceso de Descarga

Proceso: Descarga
Objetivo: Contribuir al desarrollo productivo del centro DATEC propiciando actualizaciones en cuanto a aplicaciones, herramientas y bibliografías.
Responsable: Asesor de Soporte del centro DATEC.
Ejecutores: Miembro de Proyecto, Asesor de Soporte, Especialista de Seguridad Informática.
Descripción del proceso:
1- El proceso recibe como entrada la solicitud de una descarga hecha por un

- miembro de un determinado proyecto al Asesor de Soporte del centro DATEC.
- 2- El Asesor de Soporte reenvía la solicitud de descarga al departamento de Seguridad Informática.
 - 3- El Especialista de Seguridad Informática al recibir la solicitud procede a llevar a cabo el subproceso "Realizar Descarga".
 - 4- La realización de la descarga por parte del Especialista de Seguridad Informática recibe como entrada la actividad de recepcionar la solicitud de descarga.
 - 5- Se descarga la petición, que puede ser una aplicación, una herramienta o bibliografía referente a la actualización de los proyectos.
 - 6- Se almacena el archivo descargado, siendo esta actividad el final del subproceso "Realizar descarga".
 - 7- El Especialista de Seguridad Informática notifica al Asesor de Soporte la existencia del archivo en el servidor de Seguridad Informática.
 - 8- El Asesor de Soporte accede a copiar el archivo al servidor de Seguridad Informática.
 - 9- Se almacena el archivo descargado en el servidor de descargas de DATEC.
 - 10- El Asesor de Soporte notifica al Miembro del Proyecto de la existencia del archivo solicitado en el servidor de descargas de DATEC.
 - 11- El Miembro del Proyecto descarga del servidor de DATEC el archivo solicitado, concluyendo de esta forma el proceso.

No.	Actividades	Tareas
1	Solicitar descarga	1- El Miembro del Proyecto solicita la descarga mandando un correo electrónico al Asesor de Soporte con la URL del sitio en el cual se encuentra el archivo solicitado.
2	Reenviar solicitud de descarga	1- El Asesor de Soporte reenvía la solicitud de descarga al Especialista de Seguridad Informática mandándole la URL recibida anteriormente.
3	Realizar Subproceso de descarga	1- El Especialista de Seguridad Informática procede a descargar el archivo solicitado por el Asesor de Soporte.

4	Notificar descarga	<ol style="list-style-type: none"> 1- Se notifica la descarga al Asesor de Soporte. 2- Concluye con esta actividad el subproceso "Realizar descarga".
5	Acceder al servidor de Seguridad Informática	<ol style="list-style-type: none"> 1- El Asesor de Soporte accede al servidor de Seguridad Informática para copiar el archivo solicitado.
6	Almacenar descarga en servidor DATEC	<ol style="list-style-type: none"> 1- El Asesor de Soporte almacena archivo de descarga en servidor de descargas de DATEC.
7	Notificar descarga	<ol style="list-style-type: none"> 1- El Asesor de Soporte notifica la descarga al miembro del proyecto. 2- La notificación de la descarga se realiza a través de un correo electrónico advirtiendo la disponibilidad del fichero solicitado en el servidor de descargas del centro DATEC.
8	Descargar petición	<ol style="list-style-type: none"> 1- El Miembro del Proyecto accede a descargar el archivo que solicitó directamente del servidor de descargas de DATEC. 2- En caso de que el archivo descargado presente algún problema el miembro del proyecto procede a notificar su inquietud al Asesor de Soporte y comienza nuevamente el proceso. 3- En caso de que el archivo solicitado se haya descargado sin problemas finaliza el proceso.
Subproceso "Realizar Descarga". Proceso de Descarga		
No.	Actividades	Tareas
1	Recepcionar Solicitud de Descarga	<ol style="list-style-type: none"> 1- El subproceso comienza con la recepción de la solicitud de la descarga enviada por el Asesor de Soporte del centro DATEC al Especialista de Seguridad Informática.
2	Descargar petición	<ol style="list-style-type: none"> 1- El Especialista de Seguridad Informática procede

		a descargar el fichero solicitado en la URL recibida.
3	Almacenar petición	1- Se procede a almacenar la petición materializada en un archivo en el servidor de descargas del departamento de Seguridad Informática.

2.1.3. Proceso de Protección del código fuente

La Protección del código fuente es un proceso necesario para cada una de las líneas productivas del centro DATEC, este a través de un trabajo continuo ayuda a garantizar la integridad, confidencialidad y disponibilidad de la información en cada uno de los ámbitos correspondientes al desarrollo de software. Es un proceso importante que contribuye a mantener la originalidad de cada proyecto que se lleva a cabo en cualquiera de las líneas productivas y en cuanto a Seguridad Informática presenta un alto nivel de complejidad.

Este proceso se compone de dos subprocesos secuenciales: Gestión de las Listas de Control de Acceso (ACL) y el Acceso al código por parte de los miembros del proyecto de acuerdo al rol que juegan en el desarrollo de este. Se debe especificar que el proceso de Acceso al código depende del proceso de Gestión de las ACL. Una Lista de Control de Acceso o ACL (del inglés, *Access Control List*) es un concepto de Seguridad Informática usado para fomentar la separación de privilegios. Es una forma de determinar los permisos de acceso apropiados a un determinado objeto, dependiendo de ciertos aspectos del proceso que hace el pedido.

Subproceso de Gestión de las Listas de Control de Acceso (ACL)

El subproceso de Gestión de las ACL tiene su punto de partida cuando el Administrador de la Configuración del proyecto analiza los estándares de configuración, establece los permisos en el proyecto y procede a notificar estos permisos al Administrador de Servidores. El Administrador de Servidores una vez que recibe los permisos crea las ACL y las entrega al Administrador de la Configuración el cual procede a administrarlas culminando de esta manera el subproceso de *Gestión* de las Listas de Control de Acceso (ACL).

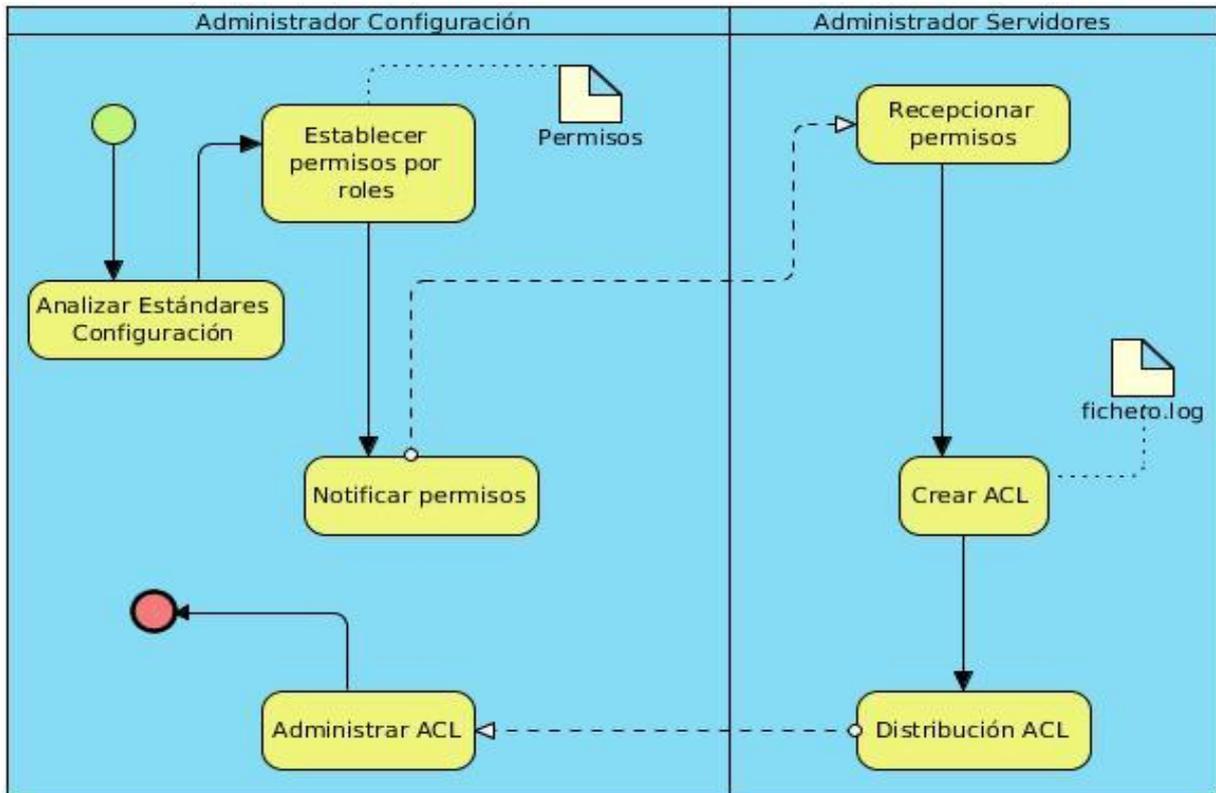


Ilustración 5. Subproceso Gestión de las Listas de Control de Acceso (ACL). Proceso de Protección del código fuente

La tabla que se muestra a continuación brinda aspectos importantes del subproceso de Gestión de las Listas de Control de Acceso (ACL), tales como: su objetivo, persona responsable y persona que lo ejecuta. Se describe de forma enumerada el proceso teniendo en cuenta todos sus posibles estados. Se enuncian todas las actividades, mencionando para cada una de estas las tareas asociadas.

Tabla 3. Descripción del subproceso de Gestión de las Listas de Control de Acceso (ACL)

<p>Proceso: Protección del código fuente. Subproceso: Gestión de las Listas de Control de Acceso (ACL).</p>
<p>Objetivos: Establecer permisos de acceso al código según las prioridades brindadas por el rol que cada miembro del proyecto pueda jugar en su desarrollo.</p>
<p>Responsable: Administrador de Servidores.</p>
<p>Ejecutores: Administrador de la Configuración del Proyecto y Administrador de Servidores.</p>

Descripción del proceso:

- 1- Analizar estándares de configuración.
- 2- Se establecen los permisos por los roles en el proyecto.
- 3- Se notifican los permisos al Administrador de Servidores.
- 4- Se recepcionan los permisos.
- 5- Crear las ACL.
- 6- Distribución de las ACL al Administrador de la Configuración del proyecto.
- 7- El Administrador de Configuración administra las ACL.

No.	Actividades	Tareas
1	Analizar estándares de configuración	1- Para la creación de los permisos el Administrador de Configuración deberá guiarse por lo orientado en el documento 5602_Estándares de Configuración.pdf . (Documento que ofrece una guía para el establecimiento de permisos al crear las ACL)
2	Establecer permisos por roles	1- El Administrador de Configuración del proyecto establece los permisos de acceso al código fuente de acuerdo a los roles correspondientes a cada uno de los módulos de cada proyecto.
3	Notificar permisos	1- El Administrador de Configuración le notifica los permisos establecidos al Administrador de Servidores.
4	Recepcionar permisos	1- El Administrador de Servidores recepciona los permisos establecidos dentro del proyecto.
5	Crear ACL	<ol style="list-style-type: none"> 1- Se crean las ACL en el servidor virtual a través de los permisos que ofrece la universidad al Administrador de Servidores. 2- Las ACL creadas en el servidor virtual se redireccionan al servidor que contiene el código fuente.

6	Distribuir ACL	1- Se distribuyen las ACL a los Administradores de Configuración en cada una de las líneas productivas del centro DATEC.
7	Administrar ACL	1- Cada Administrador de Configuración de proyecto tiene la posibilidad de realizar cambios en relación a los permisos. 2- Con esta operación finaliza el proceso.

2.1.4. Subproceso de Acceso al código

El subproceso de Acceso al código tiene su punto de partida cuando el Miembro del Proyecto accede al código. Una vez que se accede al código en caso de modificarlo se procede a realizar comentarios especificando los cambios realizados, de esta manera concluye este subproceso. En caso de que no se realicen modificaciones también se finaliza.

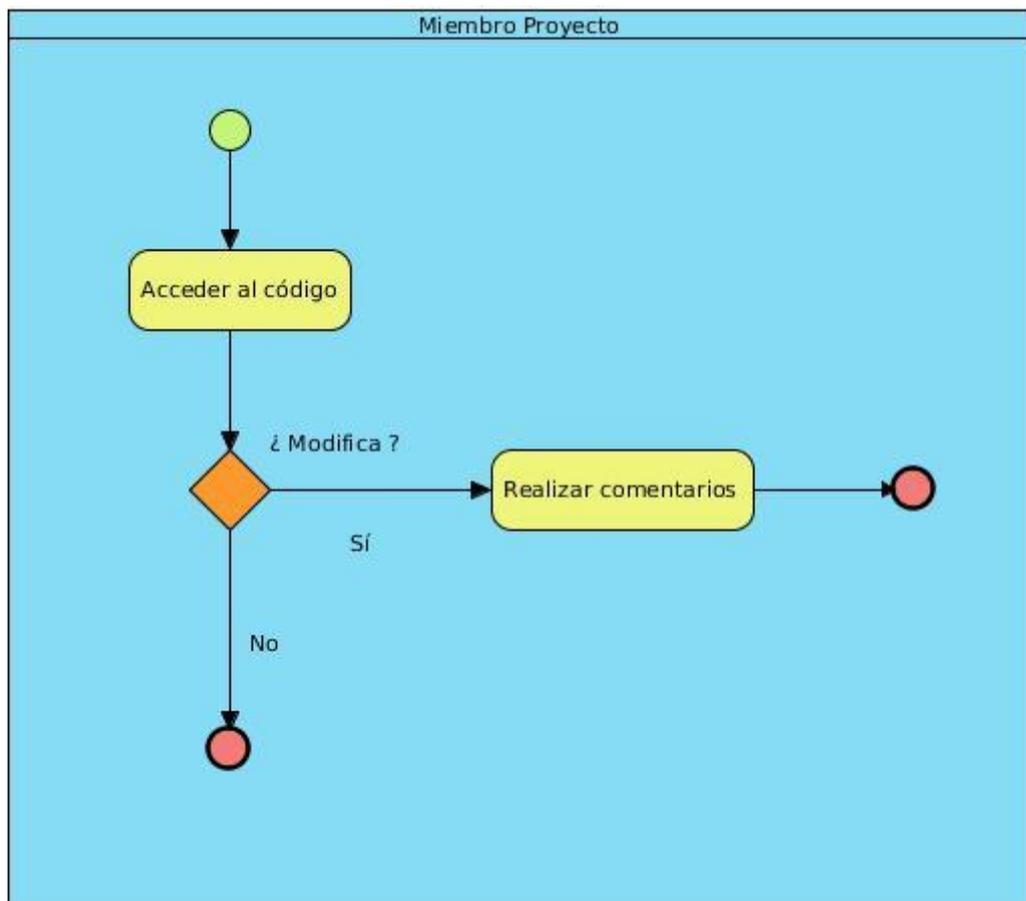


Ilustración 6. Subproceso Acceso al código. Proceso de Protección del código fuente

La tabla que se muestra a continuación brinda aspectos importantes del subproceso de Acceso al código, tales como: su objetivo, persona responsable y persona que lo ejecuta. Se describe de forma enumerada el proceso teniendo en cuenta todos sus posibles estados. Se enuncian todas las actividades, mencionando para cada una de estas las tareas asociadas.

Tabla 4. Descripción del subproceso de Acceso al código

Proceso: Protección del código fuente. Subproceso: Acceso al código.		
Objetivo: Establecer un control de los cambios realizados en el código.		
Responsable: Administrador de Servidores.		
Ejecutores: Administrador de la Configuración del proyecto y Administrador de Servidores.		
Descripción del proceso:		
<ul style="list-style-type: none"> 1- El miembro de un determinado proyecto accede con sus permisos al código. <ul style="list-style-type: none"> 1.1- En caso de modificar el código se pasa a la actividad 2 1.2- De no modificar el código se termina el proceso 2- Se procede a realizar comentarios sobre los cambios realizados y se termina el proceso. 		
No.	Actividades	Tareas
1	Acceder al código	<ul style="list-style-type: none"> 1- El Miembro del Proyecto accede al código de acuerdo a los permisos que tiene dado el rol que representa. 2- En caso de realizar alguna modificación lleva a cabo la actividad "Realizar comentarios". 3- De no realizar cambios el proceso concluye con el acceso al código.
2	Realizar comentarios	<ul style="list-style-type: none"> 1- El Miembro del Proyecto realiza un comentario por los cambios realizados previendo que si la nueva versión de código no tiene buenos resultados se pueda retroceder hacia la versión anterior. Con esta tarea finaliza la actividad y el proceso.

2.2. Establecimiento de métricas para los procesos críticos

Para el establecimiento de las métricas de la presente investigación se midieron y combinaron las actividades. Se crearon fórmulas matemáticas en vista de realizar un análisis que brinde resultados numéricos en cada uno de los procesos críticos.

2.2.1. Métricas en el proceso de Monitorización de Servidores

a) Tiempo de conexión con servidor central. Actividad: “Conectar Servidor IP”.

Esta métrica propone que la actividad “Conectar Servidor IP” se pueda medir en cuanto al tiempo de conexión (t_c en segundos), y este indicador brinda la medida de que en caso de que la conexión sea crítica ($t_c > 60$) y que no existan problemas con la red se deben tomar las medidas pertinentes en cuanto a hardware y software para que la actividad “Conectar Servidor IP” se realice de la manera más rápida posible.

Las interpretaciones de esta métrica son las siguientes:

- ✓ Cuando $t_c \leq 15$, la conexión con el servidor central es rápida.
- ✓ Cuando $15 < t_c \leq 30$, la conexión con el servidor central es lenta.
- ✓ Cuando $30 < t_c \leq 60$, la conexión con el servidor central es muy lenta.
- ✓ Cuando $t_c > 60$, la conexión con el servidor central es crítica.

b) Espacio disponible en disco duro. Actividad: “Verificar estado de servidores”.

La presente métrica propone que se lleve a cabo la verificación del estado del disco duro de los servidores. Para ello se dispone de la siguiente fórmula: $\text{Espacio Libre de Disco}(\text{eld}) \leq 0,15 * \text{Capacidad de Disco}(\text{cd})$, la cual es aplicable fundamentalmente a la partición destinada al sistema operativo, pues se debe mantener esta partición con espacio suficiente porque en caso contrario el tiempo de respuesta del sistema operativo podría reducirse considerablemente.

Las interpretaciones de esta métrica son las siguientes:

- ✓ Cuando $\text{eld} \leq 0,15 * \text{cd}$, el sistema operativo instalado en los servidores está propenso a un mal funcionamiento dependiendo del indicador eld .
- ✓ Cuando $\text{eld} > 0,15 * \text{cd}$, el sistema operativo instalado en los servidores presenta un buen rendimiento dependiendo del indicador eld .

c) Cantidad de procesos zombies. Actividad: “Verificar existencia de procesos zombies”

Esta métrica guarda una estrecha relación con la eficiencia de los servidores, los procesos zombies son aquellos que han finalizado su ejecución pero que aún no han sido eliminados de la CPU. Se recomienda que ante la existencia de este tipo de procesos se proceda a eliminarlos y para dar cumplimiento a lo sugerido anteriormente, se plantea la siguiente fórmula: Cantidad de Procesos zombies (cpz)>0.

Las interpretaciones de esta métrica son las siguientes:

- ✓ Cuando $cpz > 0$, el sistema operativo instalado en los servidores está propenso a un mal funcionamiento debido a que existen procesos zombies ocupando espacio en memoria de manera innecesaria.
- ✓ Cuando $cpz = 0$, no existen procesos zombies cargados en memoria, siendo este un aspecto favorable que tributa al buen funcionamiento del sistema operativo.

d) Porcentaje de uso del procesador. Actividad: “Verificar carga del CPU”

Un índice de uso de la CPU (Unidad Central de Procesamiento) constantemente alto puede indicar la necesidad de actualizar la CPU o de agregar varios procesadores. El uso de la CPU es el porcentaje de uso (en cuanto a ejecución de tareas de usuario o del sistema que son consideradas útiles) que tiene un procesador. Un uso elevado de la CPU puede indicar que hay una aplicación mal optimizada o diseñada. Mejorar el rendimiento de las aplicaciones que se ejecutan en los servidores del centro DATEC puede reducir el uso de la CPU. El contador Procesador: % de tiempo de procesador en el monitor de sistema es la forma más eficaz de determinar el uso de la CPU. Este contador supervisa el tiempo que la CPU dedica a la ejecución de un subproceso que no está inactivo. Para esto se recomiendan la siguientes fórmulas que indican recomendaciones de acuerdo a los posibles valores que pueda tomar el indicador Contador Procesador (cp).

Si

$50 \leq cp < 80$, entonces

Se debe de proceder a realizar revisiones en las aplicaciones que se ejecutan en el servidor en vista de verificar qué procesos consumen tanto tiempo.

Si

$cp > 80$, entonces

Se debe proceder a actualizar la CPU o bien agregar más procesadores, pues las aplicaciones que se ejecutan en el servidor demandan más recursos de hardware.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cp < 50$, la CPU del servidor presenta un uso reducido favoreciendo un buen funcionamiento de este.
- Cuando $50 \leq cp < 80$, la CPU del servidor presenta un uso elevado dependiente del consumo de las aplicaciones que se ejecutan sobre este.
- Cuando $cp > 80$, la CPU del servidor presenta un uso crítico por un excesivo consumo de las aplicaciones, demandando nuevos recursos de hardware.

e) Por ciento de consumo de la memoria RAM en cuanto a su capacidad. Actividad: “Verificar memoria RAM”

La memoria RAM juega un papel determinante para poder usar más de una aplicación en la PC o en el Sistema Operativo (SO) independientemente del SO que se use. La memoria RAM se considera como un módulo de almacenamiento temporal de datos. Es por ello que se debe medir el por ciento de uso de la RAM para tener un indicador del uso que se le está dando y de esta manera llevar a cabo acciones con los procesos que consuman demasiado espacio y tiempo en memoria. La presente métrica propone la siguiente fórmula:

$$\text{Uso de RAM (ur)} > \text{Capacidad de RAM (cr)} * 0,7$$

En caso de cumplirse esta fórmula se deben de eliminar los procesos innecesarios que presenten un consumo elevado. En el peor de los casos se debe de proceder a cambiar la memoria RAM por otra de mayor capacidad.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $ur < cr * 0,7$, la memoria RAM presenta un uso adecuado para el servidor.
- Cuando $ur > cr * 0,7$, la memoria RAM presenta un uso deficiente para el servidor.

2.2.2.Métricas en el proceso de Descarga

a) Eficiencia de Descarga. Actividades: “Solicitar descarga” y “Notificar descarga”

Para la elaboración de la presente métrica se establecieron indicadores tales como:

- Eficiencia de la descarga (E_d), se expresa en [%]
- Fecha de notificación (f_n)
- Fecha de solicitud (f_s)
- Demora (d), se expresa en [días], $d=f_n-f_s$
- Peso del archivo (p), se expresa en [MB]

Estos indicadores se relacionan en la siguiente fórmula: $E_d = (1-d/p)*100$, debe destacarse que en caso de que el indicador Demora (d), tome el valor numérico 0, es decir $d=0$, automáticamente d tomará valor 1, por tanto $d=1$ debido a que el indicador d estará en el rango de un solo día tomando cualquiera de estos dos valores.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $E_d < 50\%$, la descarga realizada se clasifica como Deficiente.
- Cuando $50\% \leq E_d < 90\%$, la descarga realizada se clasifica como Adecuada.
- Cuando $E_d \geq 90\%$, la descarga realizada se clasifica como Eficiente.

b) Peso del archivo descargado. Subproceso: "Realizar Descarga". Actividad: "Descargar Petición".

La presente métrica tiene como objetivo establecer tres tipos de clasificaciones para una descarga realizada atendiendo al peso del archivo descargado, estas clasificaciones variarán de acuerdo a los valores que tome el indicador Peso del Archivo (p).

Las interpretaciones de esta métrica son las siguientes:

- Cuando $p < 1$ MB, la descarga se clasifica en simple.
- Cuando $1 \text{ MB} \leq p < 1024$ MB, la descarga se clasifica en relevante.
- Cuando $p \geq 1024$ MB, la descarga se clasifica en crítica.

c) Tiempo de recepción de la descarga. Actividad: "Solicitar descarga". Subproceso: "Realizar Descarga". Actividad: "Recepcionar Descarga".

La relación que se establece en la presente métrica con estas actividades es para determinar el tiempo en horas que transcurre desde que un miembro de un proyecto solicita la descarga y esta es recepcionada por el Especialista de Seguridad Informática.

Se establecieron indicadores tales como:

- Tiempo de recepción (tr), se expresa en [horas]
- Inicio de Descarga (id), toma valores cualitativos como [Adecuada o Deficiente]
- Fecha de Recepcionar Descarga (fr), se expresa en formato de fecha
- Fecha de Solicitar descarga (fs), se expresa en formato de fecha

El indicador tr se calcula de la siguiente manera: $tr = fr - fs$.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $tr \leq 24$, el indicador id se considera Adecuado.
- Cuando $tr > 24$, el indicador id se considera Deficiente.

d) Por ciento de integridad del Archivo. Actividad: "Descargar Petición".

Esta actividad está presente en dos de las piscinas del proceso de Descarga, en la piscina de Miembro de Proyecto y en la de Especialista de Seguridad Informática específicamente dentro del subproceso: "Realizar Descarga". Es importante establecer un indicador capaz de medir la integridad del archivo descargado primeramente por el Especialista de Seguridad Informática y luego por el Miembro del Proyecto que accede al servidor de descargas del centro DATEC. Para esto se toma como referencia el peso del archivo descargado.

Para la elaboración de la presente métrica se establecieron indicadores tales como:

- Por ciento de integridad (pin), se expresa en [%]
- Peso Inicial (pi), se expresa en [MB]
- Peso final (pf), se expresa en [MB]

El indicador pi hace referencia al peso del archivo que se almacena en el servidor de descargas del centro DATEC. Por su parte el indicador pf hace referencia al peso del archivo cuando llega a manos del Miembro del Proyecto.

Los indicadores mencionados se relacionan en la siguiente fórmula: $pin = (pf/pi) * 100$

Las interpretaciones de esta métrica son las siguientes:

- Cuando $pin \neq 100$, el archivo descargado por el Miembro del Proyecto sufrió diversos cambios o simplemente no es el archivo que se descargó de acuerdo a la solicitud enviada.
- Cuando $pin = 100$, el archivo descargado coincide al menos en cuanto al peso.

e) Espacio disponible en servidor de descargas del centro DATEC. Actividad: "Almacenar Descarga en servidor DATEC".

La presente métrica hace referencia a la actividad "Almacenar Descarga en servidor DATEC", la cual es llevada a cabo por el Asesor de Soporte, la misma tiene en cuenta el peso del archivo descargado y el espacio disponible en el servidor de descargas del centro DATEC. Para ello se definen indicadores tales como:

- Peso de Archivo (p), se expresa en [MB]
- Espacio disponible en servidor de descargas DATEC (ed), se expresa en [MB]

Las interpretaciones de esta métrica son las siguientes:

- Cuando $p > ed$, el Asesor de Soporte debe a proceder a liberar espacio en el servidor de descargas del centro DATEC.
- Cuando $p = ed$, se recomienda en caso de ser posible liberar espacio en el servidor de descargas del centro DATEC.
- Cuando $p < ed$, el servidor de descargas del centro DATEC se encuentra en buenas condiciones para llevar a cabo el almacenamiento del archivo descargado.

f) Rapidez de la descarga. Actividades: "Solicitar descarga" y "Notificar descarga".

Para la realización de la presente métrica se establecieron indicadores tales como:

- Nivel de Rapidez (nr), se expresa en [días]
- Fecha de Inicio (fi), se expresa en formato de fecha

- Fecha de cumplimiento (fc), se expresa en formato de fecha

La variable fc hace referencia a la fecha en la que el Asesor de Soporte ya ha almacenado el archivo solicitado por el Miembro del Proyecto y procede a notificarle a este a través de un correo que la descarga se encuentra disponible en el servidor de descargas del centro DATEC. Los indicadores mencionados anteriormente se relacionan en la siguiente fórmula: $nr=fc-fi$.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $nr \leq 7$ días, la descarga se clasifica en Descarga Rápida.
- Cuando $7 < nr \leq 14$ días, la descarga se clasifica en Descarga Lenta.
- Cuando $nr > 14$ días, la descarga se clasifica en Descarga Muy lenta.

2.2.3. Métricas en el subproceso de Gestión de las Listas de Control de Acceso (ACL). Proceso de Protección del código fuente.

a) Cantidad de permisos de acuerdo a roles. Actividad: “Establecer permisos por roles”

La presente métrica es elaborada con motivo de aumentar el control a la hora de establecer los permisos en las distintas líneas productivas del centro DATEC, para ello se establecen indicadores tales como:

- Cantidad de Miembros de las ACL (cm)
- Matrícula total del proyecto (mp)

Los indicadores antes mencionados se relacionan en la siguiente fórmula: $cm > 0,5 * mp$. En caso de cumplirse esta fórmula entonces se debe de considerar un nuevo análisis al establecer permisos en vista que solo tengan acceso al código fuente el número de usuarios estrictamente necesarios.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cm > 0,5 * mp$, existen demasiadas ACL en el proyecto y esto puede ser un factor crítico para la seguridad del código fuente.
- Cuando $cm \leq 0,5 * mp$, el número de ACL creadas es limitado siendo este un aspecto favorable para la seguridad del código fuente.

b) Coincidencia en cuanto a cantidad de permisos. Actividades: “Establecer permisos por roles” y “Recepcionar permisos”.

La presente métrica se establece a partir de la relación existente entre estas dos actividades en vista de chequear que la cantidad de permisos establecidos previamente por el Administrador de Configuración del proyecto son recepcionados correctamente por el Administrador de Servidores, para ello se crean indicadores tales como:

- Cantidad de permisos establecidos (ce)
- Cantidad de permisos recepcionados (cr)

Los indicadores definidos se relacionan en la siguiente fórmula: $ce \neq cr$, en caso de cumplirse el Administrador de Configuración del proyecto y el Administrador de Servidores deben de rectificar para que la cantidad de permisos establecidos por el primero sean recepcionados correctamente por el segundo.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $ce \neq cr$, existe descoordinación entre las partes responsables de establecer y recepcionar los permisos para llevar a cabo la creación de las ACL.
- Cuando $ce = cr$, existe una adecuada coordinación en la creación de permisos para el establecimiento de las ACL.

c) Coincidencia de permisos recepcionados con respecto a la cantidad de permisos dentro de las ACL. Actividades: “Recepcionar Permisos” y “Crear ACL”.

La recepción de permisos por parte del Administrador de Servidores es el punto de partida para la creación de las ACL. Para esto es necesario tener en cuenta que la cantidad de permisos recepcionados debe de coincidir con la cantidad de permisos establecidos en cada una de las ACL. Para la realización de la presente métrica se establecieron indicadores tales como:

- Cantidad permisos recepcionados (cr)
- Cantidad permisos dentro de la ACL (cd)

Estos indicadores se relacionan a través de la siguiente fórmula: $cr \neq cd$, en caso de cumplirse el Administrador de Servidores debe tomar las medidas para hacer coincidir estos permisos en cuanto a la cantidad contenida dentro de la ACL.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cr \neq cd$, las ACL creadas por el Administrador de Servidores carecen de similitud con respecto a la cantidad de permisos establecidos por el Administrador de Configuración.
- Cuando $cr = cd$, las ACL creadas por el Administrador de Servidores coinciden en cantidad de permisos con lo establecido por el Administrador de Configuración.

d) Cantidad de ACL distribuidas con respecto a la cantidad de ACL creadas. Actividades: “Crear ACL” y “Distribución ACL”

El Administrador de Servidores se encuentra en el deber de distribuir a los Administradores de Configuración de cada una de las líneas productivas las ACL creadas. Para la elaboración de la presente métrica se establecieron indicadores tales como:

- Cantidad de ACL creadas (cc)
- Cantidad de ACL distribuidas (cd)

Estos indicadores se relacionan en la siguiente fórmula: $cc \neq cd$, en caso de cumplirse, el Administrador de Servidores deberá encargarse de distribuir todas las ACL creadas procurando que exista igualdad entre las ACL creadas y las ACL distribuidas.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cc \neq cd$, existe descoordinación al distribuir las ACL creadas al menos con respecto a la cantidad.
- Cuando $cc = cd$, la distribución de las ACL creadas coincide al menos en número con las distribuidas.

e) Cantidad de ACL distribuidas con respecto a la cantidad de ACL administradas. Actividades: “Distribución ACL” y “Administrar ACL”.

El Administrador de Servidores debe de distribuir cada una de las ACL que el mismo cree, pero el Administrador de Configuración de proyecto está en el deber de administrar cada una de las ACL que le son distribuidas. Para la creación de la presente métrica se establecieron indicadores tales como:

- Cantidad ACL distribuidas (cd)
- Cantidad ACL administradas (ca)

Estos indicadores se relacionan en la siguiente fórmula: $cd \neq ca$, que en caso de cumplirse el Administrador de Configuración debe de proceder a administrar sus ACL correspondientes y asumir con toda responsabilidad los retos respecto a las tareas asociadas.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cd \neq ca$, existe descoordinación entre la cantidad de ACL distribuidas al Administrador de Configuración y las que este administra en realidad.
- Cuando $cd = ca$, el Administrador de Configuración administra la misma cantidad de ACL que le han sido distribuidas siendo esto un aspecto favorable para el proceso.

2.2.4. Métricas en el subproceso de Acceso al código. Proceso de Protección del código fuente

a) Factor de coincidencia de cambios con comentarios. Actividad: “Realizar comentarios”.

La presente métrica permite a los desarrolladores tener el control sobre los cambios realizados para que en caso de que las nuevas versiones de código fuente presenten problemas se pueda llevar a cabo un retroceso hacia las versiones anteriores. Para la creación de esta métrica se establecieron indicadores tales como:

- Factor de coincidencia (fc)
- Cantidad de modificaciones (cm)
- Cantidad Comentarios (cc)

Estos indicadores se relacionan en la siguiente fórmula: $fc = cm - cc$.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $fc > 0$, existe disparidad en cuanto a la cantidad de modificaciones llevadas a cabo en el código fuente y los comentarios realizados sobre estas modificaciones.
- Cuando $fc < 0$, existe un excesivo número de comentarios y por tanto se recomienda simplificarlos en vista que haya al menos un comentario por cada cambio realizado.
- Cuando $fc = 0$, la cantidad de comentarios coincide con la cantidad de modificaciones llevadas a cabo en el código fuente.

b) Coincidencia de fechas. Actividades: “Acceder a código” y “Realizar comentarios”.

En caso de que un miembro de un proyecto acceda al código y realice cambios en una determinada fecha se debe de verificar que el comentario tenga la fecha del día en que se realizó el cambio. Para la elaboración de la presente métrica se establecieron indicadores tales como:

- Fecha de Modificación (fm)
- Fecha de Comentario (fc)

Estos indicadores se relacionan en la siguiente fórmula: $fm \neq fc$, que en caso de cumplirse se debe corregir de forma inmediata debido a que es necesario que el cambio realizado en un segmento de código en una fecha específica esté comentariado con la misma fecha.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $fm \neq fc$, existe descoordinación entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha.
- Cuando $fm = fc$, existe una coordinación adecuada entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha.

c) Tipo de Modificación con respecto a la cantidad de líneas de código. Actividad: “Acceder al código”

Para la elaboración de la presente métrica se establecen indicadores tales como:

- Tipo de Modificación (tm)
- Cantidad líneas de código antes (cla)
- Cantidad líneas de código después (cld)

Las distintas clasificaciones que puede tener el indicador tm brindan una medida de la relación existente entre la cantidad de líneas anteriores y posteriores a la modificación dentro del código.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cla < cld$, ocurre un tm por agregación.
- Cuando $cla = cld$, ocurre un tm por sustitución.
- Cuando $cla > cld$, ocurre un tm por disminución.

d) Cantidad de accesos por día. Actividad: “Acceder al código”.

Para la creación de la presente métrica se establece el indicador:

- Accesos por día (ad)

Este indicador brinda la medida de la cantidad de veces que accede un solo miembro del proyecto al código fuente y se plantea en la siguiente fórmula: $ad \geq 5$, que en caso de cumplirse el Administrador de Configuración de proyecto debe de proceder a realizar un análisis para saber el por qué del acceso de forma reiterada de ese miembro del proyecto al código fuente.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $ad \geq 5$, existe un excesivo acceso al código fuente por parte de ese determinado miembro del proyecto.
- Cuando $ad < 5$, existe un acceso al código fuente adecuado por parte de ese determinado miembro del proyecto.

e) Cantidad de Modificaciones al código. Actividad: “Acceder al código”.

La presente métrica se establece debido a que si un determinado usuario accede al código, y pretende realizar modificaciones en este, se recomienda hacer un análisis previo para que en un mismo día realice todas las modificaciones necesarias en la menor cantidad de accesos posibles. Para la elaboración de esta métrica se estableció el indicador:

- Cantidad de Modificaciones por día (cmd)

Este indicador se pone de manifiesto en la siguiente fórmula: $cmd \geq 3$. En caso de cumplirse el Administrador de Configuración de proyecto debe de llamar la atención de ese usuario en vista que los cambios que se deseen hacer en el código se analicen bien y en el mejor de los casos se hagan con un solo acceso al código fuente. Es decir cuando $cmd = 1$, que es lo más recomendable.

Las interpretaciones de esta métrica son las siguientes:

- Cuando $cmd \geq 3$, existe un excesivo número de modificaciones por parte de un usuario en el código fuente.
- Cuando $cmd < 3$, existe un adecuado número de modificaciones por parte de un usuario en el código fuente.
- Cuando $cmd = 1$, es el número de modificaciones al código fuente más apropiado para un usuario.

2.3. Métricas generales

La tabla que se muestra a continuación tiene un resumen de cada una de las métricas elaboradas en el presente capítulo, siendo estas un total de veintiuna métricas para los procesos críticos del centro DATEC.

Tabla 5. Métricas generales

Proceso de Monitorización de servidores
a) Tiempo de conexión con servidor central. Actividad: "Conectar Servidor IP".
b) Espacio disponible en disco duro. Actividad: "Verificar estado de servidores".
c) Cantidad de procesos zombies. Actividad: "Verificar existencia de procesos zombis"
d) Porcentaje de uso del procesador. Actividad: "Verificar carga del CPU"
e) Por ciento de consumo de la memoria RAM en cuanto a su capacidad. Actividad: "Verificar memoria RAM"
Proceso de Descarga
a) Eficiencia de Descarga. Actividades: "Solicitar descarga" y "Notificar descarga" (Desde la piscina "Asesor de Soporte")
b) Peso del archivo descargado. Subproceso: "Realizar Descarga". Actividad: "Descargar Petición".
c) Tiempo de recepción de la descarga. Actividad: "Solicitar descarga". Subproceso: "Realizar Descarga". Actividad: "Recepcionar Descarga".
d) Por ciento de integridad del Archivo. Actividad: "Descargar Petición".
e) Espacio disponible en servidor de descargas del centro DATEC. Actividad: "Almacenar Descarga en servidor DATEC".
f) Rapidez de la descarga. Actividades: "Solicitar descarga" y "Notificar descarga".
Proceso de Protección del código fuente. Subproceso de Gestión de Listas de Control de Acceso (ACL)
a) Cantidad de permisos de acuerdo a roles. Actividad: "Establecer permisos por roles"
b) Coincidencia en cuanto a cantidad de permisos. Actividades: "Establecer permisos por roles" y "Recepcionar permisos".

c) Coincidencia de permisos recepcionados con respecto a la cantidad de permisos dentro de las ACL. Actividades: “Recepcionar Permisos” y “Crear ACL”.
d) Cantidad de ACL distribuidas con respecto a la cantidad de ACL creadas. Actividades: “Crear ACL” y “Distribución ACL”
e) Cantidad de ACL distribuidas con respecto a la cantidad de ACL administradas. Actividades: “Distribución ACL” y “Administrar ACL”.
Proceso de Protección del código fuente. Subproceso de Acceso al código
a) Factor de coincidencia de cambios con comentarios. Actividad: “Realizar comentarios”.
b) Coincidencia de fechas. Actividades: “Acceder a código” y “Realizar comentarios”.
c) Tipo de Modificación con respecto a la cantidad de líneas de código. Actividad: “Acceder al código”.
d) Cantidad de accesos por día. Actividad: “Acceder al código”.
e) Cantidad de Modificaciones al código. : “Acceder al código”.

Conclusiones del capítulo

En el presente capítulo se llevó a cabo el modelado y descripción de los procesos críticos de soporte del centro DATEC. Se diseñaron métricas sobre la base de lo descrito en la norma ISO 14764. Para su elaboración se midieron las actividades de los procesos relacionadas con los indicadores definidos.

Capítulo 3. Aplicación de métricas

Introducción

En el presente capítulo se aplican las métricas propuestas a través de la simulación estableciendo valores probables. Se muestran los resultados de las mediciones realizadas en un entorno real a las métricas propuestas durante un período de 45 días hábiles en el centro DATEC.

3.1. Simulación

La simulación es la representación de uno o más procesos que permite analizar sus características dentro de un entorno ficticio. Permite obtener estimaciones sobre el comportamiento de los procesos de acuerdo a los parámetros usados en las entradas e intenta reproducir la realidad a partir de la resolución numérica. Cuando se simula un proceso, se pueden medir cada una de sus actividades, y obtener una visión de la eficacia de este a través de los diversos valores que puedan tomar los indicadores bajo circunstancias propiamente definidas. La simulación de los procesos críticos del grupo de soporte del centro DATEC se lleva a cabo en este trabajo de diploma con el propósito de entender mejor su comportamiento, y poder evaluar diversas estrategias que permitan una mejor operatividad.

Para llevar a cabo la simulación de los procesos críticos de soporte del centro DATEC se utilizó la herramienta Bonita Open Solution en su versión 5.6.2. Es importante destacar que este proceso brinda resultados probabilísticos teniendo en cuenta los siguientes aspectos:

- Cantidad de días
- Cantidad de horas
- Cantidad de instancias simuladas
- Cantidad de compuertas o bifurcaciones

Los resultados se ofrecerán en la unidad de medida *por ciento* (%).

Imágenes del proceso de simulación

Las imágenes que se muestran a continuación brindan una descripción del proceso de simulación llevado a cabo en la herramienta Bonita Open Solution. Cada una hace referencia al comportamiento de una métrica dentro de una actividad de proceso.

La siguiente imagen hace referencia a la simulación de la métrica Espacio disponible en disco perteneciente al proceso de Monitorización de Servidores. Se simularon 100 instancias durante 2 semanas para un período de 8 horas diarias.

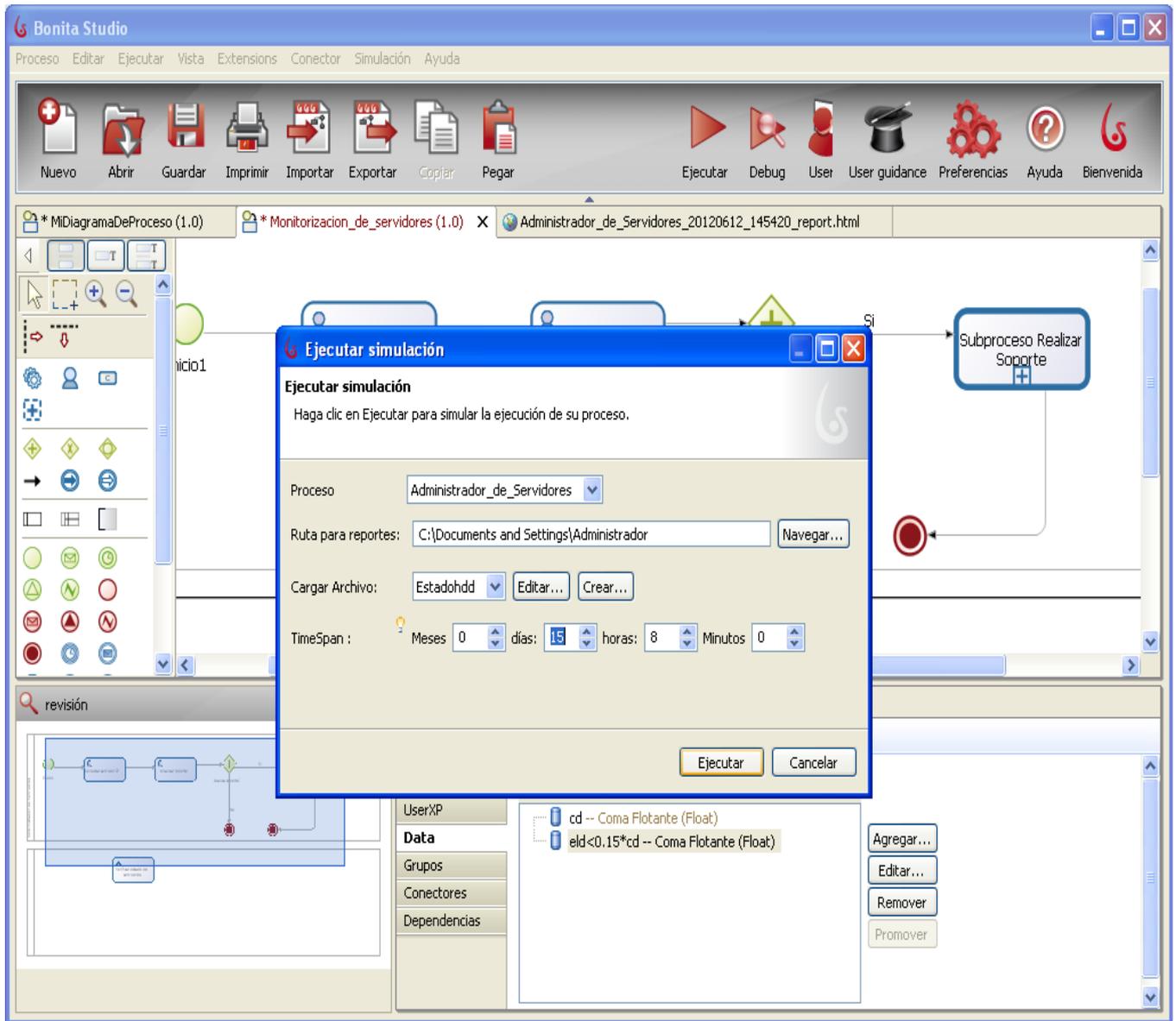


Ilustración 7. Métrica: Espacio disponible en disco duro. Proceso de Monitorización de Servidores.

La presente imagen hace referencia a la simulación de la métrica Rapidez de la Descarga perteneciente al proceso de Descarga. Se simularon 100 instancias durante 2 semanas para un período de 8 horas diarias.

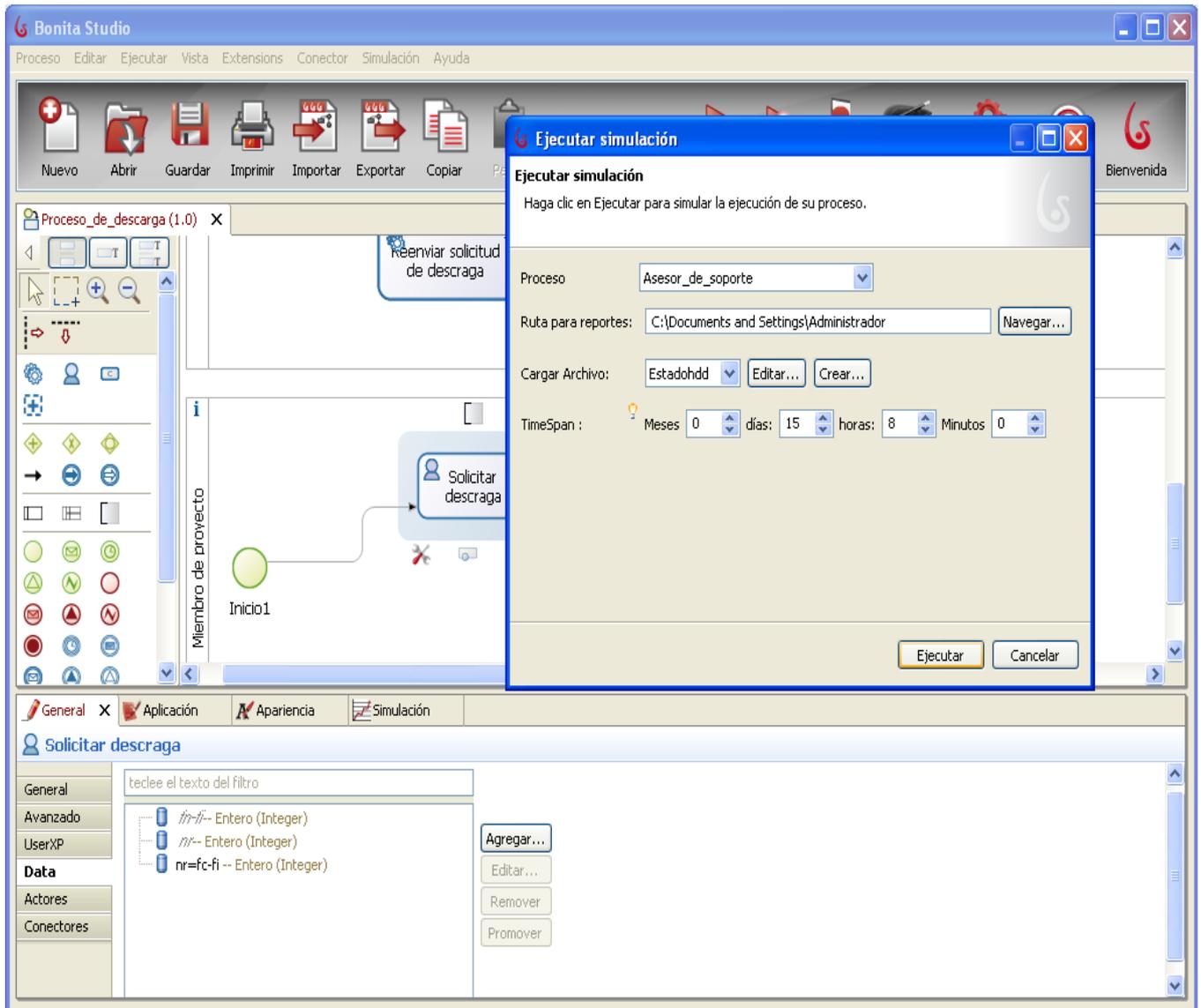


Ilustración 8. Métrica: Rapidez de la Descarga. Proceso de Descarga.

La presente imagen hace referencia a la simulación de la métrica Cantidad de permisos de acuerdo a roles perteneciente al proceso de Protección del código fuente. Se simularon 100 instancias durante 2 semanas para un período de 8 horas diarias.

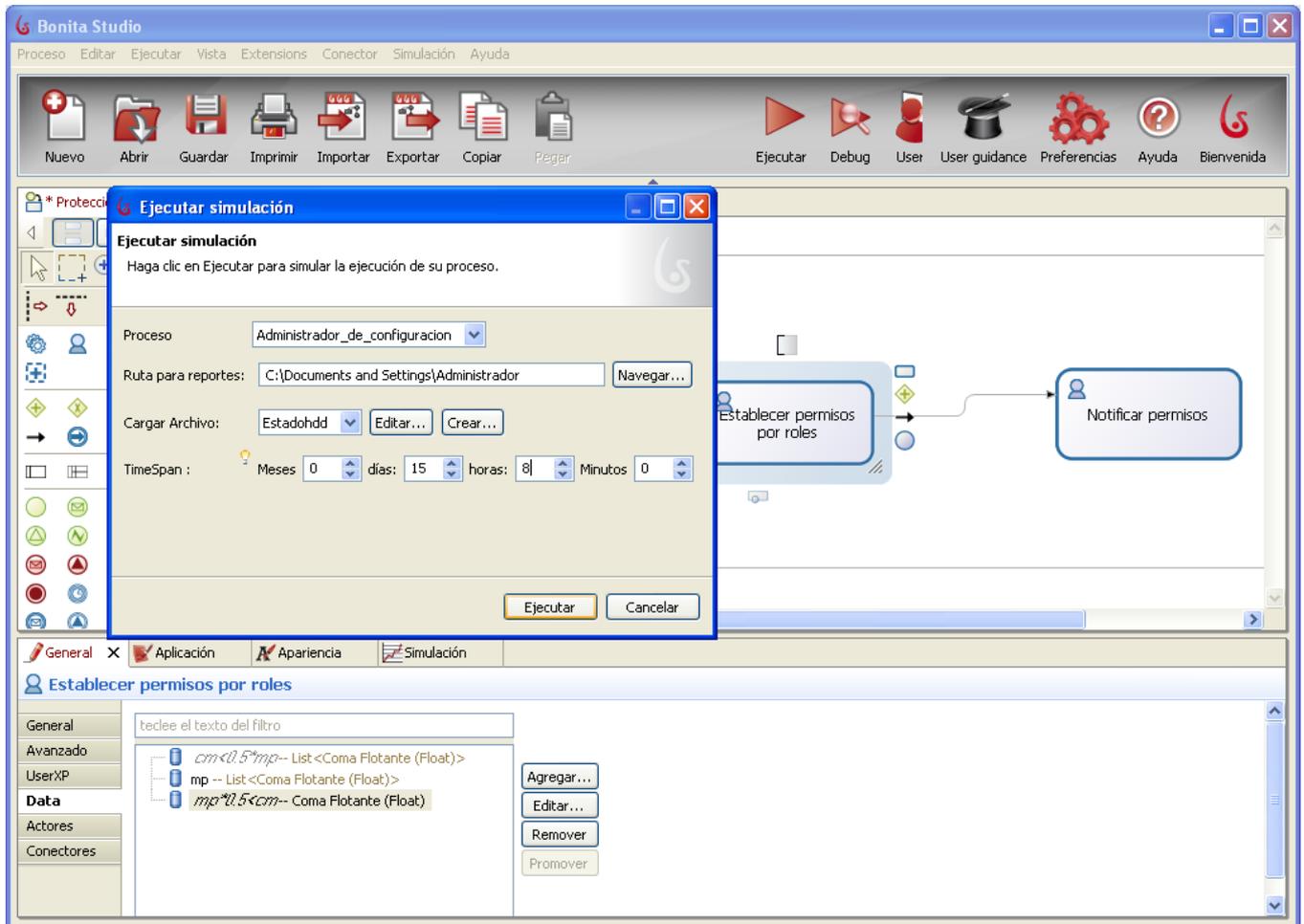


Ilustración 9. Métrica: Cantidad de permisos de acuerdo a Roles. Proceso de Protección del código fuente.

3.1.1. Simulación del proceso de “Monitorización de Servidores”

- Cantidad de días: 25
- Cantidad de horas: 200
- Cantidad de instancias simuladas: 100
- Cantidad de compuertas o bifurcaciones: 5

Compuerta "¿Operaciones de soporte?": 21% si accede, 79% no accede

Compuerta "¿HDD lleno?": 35% si accede, 65% no accede

Compuerta "¿Estado crítico?": 40% si accede, 60% no accede

Compuerta "¿Existen actualizaciones?": 75% si accede, 25% no accede

Compuerta "¿Existen procesos zombis?": 7% si accede, 93% no accede

3.1.2. Comportamiento de los indicadores durante la simulación del proceso de Monitorización de servidores

- ✓ Durante la simulación de la métrica Tiempo de conexión con servidor central la variable Tiempo de conexión (tc) tuvo diversos comportamientos tales como:
 - $tc \leq 15$, siendo una conexión rápida en el 68% de las instancias simuladas.
 - $15 < tc \leq 30$, siendo una conexión lenta en el 13% de las instancias simuladas.
 - $30 < tc \leq 60$, siendo una conexión muy lenta en el 9% de las instancias simuladas.
 - $tc > 60$, siendo una conexión crítica en el 10% de las instancias simuladas.

- ✓ Durante la simulación de la métrica Espacio disponible en disco duro las variables Espacio Libre de Disco (eld) y Capacidad de Disco (cd) tuvieron diversos comportamientos dependiendo de su relación como se muestra a continuación:
 - $eld \leq 0,15 * cd$, siendo un factor desfavorable para el funcionamiento del sistema operativo en el 65% de las instancias simuladas.
 - $eld > 0,15 * cd$, siendo un factor favorable para el funcionamiento del sistema operativo en el 35% de las instancias simuladas.

- ✓ Durante la simulación de la métrica Cantidad de procesos zombies la variable Cantidad de Procesos zombies (cpz) tuvo comportamientos tales como:
 - $cpz > 0$, siendo este un factor desfavorable para el funcionamiento del sistema operativo en el 7% de las instancias simuladas.
 - $cpz = 0$, un este un factor favorable para el funcionamiento del sistema operativo en el 93% de las instancias simuladas.

- ✓ Durante la simulación de la métrica Porcentaje de uso del procesador la variable Contador Procesador (cp) tuvo comportamientos tales como:
 - $cp < 50$, presentando la CPU un uso reducido en el 17% de las instancias simuladas.
 - $50 \leq cp < 80$, presentando la CPU un uso elevado en el 32% de las instancias simuladas.

- $cp > 80$, presentando la CPU un uso crítico en el 51% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Por ciento de consumo de la memoria RAM en cuanto a su capacidad las variables Uso de RAM (ur) y Capacidad de RAM (cr) tuvieron comportamientos tales como:
 - $ur < cr * 0,7$, presentando la RAM un uso adecuado para el servidor en el 97% de las instancias simuladas.
 - $ur > cr * 0,7$, presentando la RAM un uso deficiente para el servidor en el 13% de las instancias simuladas.

3.1.3. Simulación del proceso de Descarga

- Cantidad de días: 25
- Cantidad de horas: 200
- Cantidad de instancias simuladas: 100
- Cantidad de compuertas o bifurcaciones: 0

3.1.4. Comportamiento de los indicadores durante la simulación del proceso de Descarga

- ✓ Durante la simulación de la métrica Eficiencia de Descarga la relación de los indicadores Eficiencia de la descarga (Ed), Fecha de notificación (fn), Fecha de solicitud (fs), Demora (d), Peso del archivo (p) se evidenció a través de la siguiente fórmula: $Ed = (1 - d/p) * 100$, teniendo el indicador Ed comportamientos tales como:
 - $Ed < 50\%$, siendo la descarga deficiente en el 41% de las instancias simuladas.
 - $50\% \leq Ed < 90\%$, siendo la descarga adecuada en el 56% de las instancias simuladas.
 - $Ed \geq 90\%$, siendo la descarga eficiente en el 3% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Peso del archivo descargado el indicador Peso del Archivo (p) presentó comportamientos tales como:
 - $p < 1$ MB, siendo la descarga simple en el 35% de las instancias simuladas.
 - $1\text{MB} \leq p < 1024$ MB, siendo la descarga relevante en el 55% de las instancias simuladas.

- $p \geq 1024$ MB, siendo la descarga crítica en el 10% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Tiempo de recepción de la descarga la relación de los indicadores Tiempo de recepción (tr), Inicio de Descarga (id), Fecha de Recepcionar Descarga (fr), Fecha de Solicitar descarga (fs), se evidenció en la siguiente fórmula: $tr = fr - fs$, donde el indicador id tomo valores cualitativos de acuerdo a los valores cuantitativos de tr.
- $tr \leq 24$, siendo id adecuado en el 73% de las instancias simuladas.
- $tr > 24$, siendo id deficiente en el 17% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Por ciento de integridad del Archivo la relación de los indicadores Por ciento de integridad (pin), Peso Inicial (pi), Peso final (pf) se evidenció en la siguiente fórmula: $pin = (pf/pi) * 100$, donde el indicador pin tuvo comportamientos tales como:
 - $pin \neq 100$, no siendo el archivo descargado el solicitado en el 50% de las instancias simuladas.
 - $pin = 100$, siendo el archivo descargado el solicitado en el 50% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Espacio disponible en servidor de descargas del centro DATEC la relación de los indicadores Peso de Archivo (p) y Espacio disponible en servidor de descargas DATEC (ed) tuvo comportamientos tales como:
 - $p > ed$, se debe proceder a liberar espacio en el servidor según el 8% de las instancias simuladas.
 - $p = ed$, no es obligado pero se recomienda liberar espacio en el servidor en un 0% según las instancias simuladas.
 - $p < ed$, el servidor de descargas del centro DATEC se encuentra en buenas condiciones para llevar a cabo el almacenamiento del archivo descargado según el 92% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Rapidez de la descarga la relación entre los indicadores Nivel de Rapidez (nr), Fecha de Inicio (fi), Fecha de cumplimiento (fc) se evidencia a través de la fórmula $nr = fc - fi$, teniendo nr comportamientos tales como:
 - $nr \leq 7$, siendo una descarga rápida en el 22% de las instancias simuladas.
 - $7 < nr \leq 14$, siendo una descarga lenta el 71% de las instancias simuladas.

- $nr > 14$, siendo una descarga muy lenta el 7% de las instancias simuladas.

3.1.5. Simulación del subproceso de Gestión de las Listas de Control de Acceso (ACL). Proceso de protección del código fuente

- Cantidad de días: 25
- Cantidad de horas: 200
- Cantidad de instancias simuladas: 100
- Cantidad de compuertas o bifurcaciones: 0

3.1.6. Comportamiento de los indicadores durante la simulación del subproceso de Gestión de las ACL.

- ✓ Durante la simulación de la métrica Cantidad de permisos de acuerdo a roles la relación entre los indicadores Cantidad de Miembros de las ACL (cm) y Matrícula total del proyecto (mp) se evidencia a través de la fórmula: $cm > 0,5 * mp$, presentando comportamientos tales como:
 - $cm > 0,5 * mp$, se manifiesta una amplia existencia de permisos en el proyecto siendo un caso desfavorable en el 81% de las instancias simuladas.
 - $cm \leq 0,5 * mp$, la existencia de permisos en el proyecto es limitada siendo un caso favorable según el 19% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Coincidencia en cuanto a cantidad de permisos la relación entre los indicadores Cantidad de permisos establecidos (ce) y Cantidad de permisos recepcionados (cr) presentó comportamientos tales como:
 - $ce \neq cr$, existiendo descoordinación entre las partes responsables de establecer y recepcionar los permisos para la creación de las ACL en el 23% de las instancias simuladas.
 - $ce = cr$, existiendo buena coordinación entre las partes responsables de establecer y recepcionar los permisos para la creación de las ACL en el 77% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Coincidencia de permisos recepcionados con respecto a la cantidad de permisos dentro de las ACL la relación entre los indicadores Cantidad permisos recepcionados (cr) y Cantidad permisos dentro de la ACL (cd) presentó comportamientos tales como:

- $cr \neq cd$, las ACL creadas por el Administrador de Servidores carecen de similitud con respecto a la cantidad de permisos establecidos por el Administrador de Configuración según el 60% de las instancias simuladas.
 - $cr = cd$, las ACL creadas por el Administrador de Servidores coinciden en cantidad de permisos con lo establecido por el Administrador de Configuración según el 40% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Cantidad de ACL distribuidas con respecto a la cantidad de ACL creadas la relación entre los indicadores Cantidad de ACL creadas (cc) y Cantidad de ACL distribuidas (cd) presentó comportamientos tales como:
- $cc \neq cd$, existe descoordinación al distribuir las ACL creadas al menos con respecto a la cantidad en el 49% de las instancias simuladas.
 - $cc = cd$, la distribución de las ACL creadas coincide al menos en número con las distribuidas en el 51% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Cantidad de ACL distribuidas con respecto a la cantidad de ACL administradas la relación entre los indicadores Cantidad ACL distribuidas (cd) y Cantidad ACL administradas (ca) presentó comportamientos tales como:
- $cd \neq ca$, existe descoordinación entre la cantidad de ACL distribuidas al Administrador de Configuración y las que este administra en realidad según el 47% de las instancias simuladas.
 - $cd = ca$, el Administrador de Configuración administra al menos la misma cantidad de ACL que le han sido distribuidas siendo esto un aspecto favorable en el proceso de Gestión de las ACL según el 53% de las instancias simuladas.

3.1.7. Simulación del subproceso de Acceso al código. Proceso de protección del código fuente.

- Cantidad de días: 25
- Cantidad de horas: 200
- Cantidad de instancias simuladas: 100
- Cantidad de compuertas o bifurcaciones: 1
- Compuerta "¿Modifica?": 80% si accede, 20% no accede

3.1.8. Comportamiento de los indicadores durante la simulación del subproceso de Acceso al código.

- ✓ Durante la simulación de la métrica Factor de coincidencia de cambios con comentarios la relación entre los indicadores Factor de coincidencia (fc), Cantidad de modificaciones (cm) y Cantidad comentarios (cc) se evidencia a través de la fórmula: $fc=cm-cc$, presentando comportamientos tales como:
 - $fc>0$, existe disparidad en cuanto a la cantidad de modificaciones llevadas a cabo en el código fuente y los comentarios realizados sobre estas modificaciones en el 37% de las instancias simuladas.
 - $fc<0$, existe un excesivo número de comentarios y por tanto se recomienda simplificarlos en vista que haya al menos un comentario por cada cambio realizado 32% de las instancias simuladas.
 - $fc=0$, la cantidad de comentarios coincide con la cantidad de modificaciones llevadas a cabo en el código fuente 31% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Coincidencia de fechas la relación entre los indicadores Fecha de Modificación (fm) y Fecha de Comentario (fc) presentó comportamientos tales como:
 - $fm\neq fc$, existe descoordinación entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha en el 25% de las instancias simuladas.
 - $fm=fc$, existe una coordinación adecuada entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha en el 75% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Tipo de Modificación con respecto a la cantidad de líneas de código la relación entre los indicadores Tipo de Modificación (tm), Cantidad líneas de código antes (cla) y Cantidad líneas de código después (cld) presentó comportamientos tales como:
 - $cla<cld$, ocurrió un tm por agregación en el 30% de las instancias simuladas.
 - $cla=cld$, ocurrió un tm por sustitución en el 45% de las instancias simuladas.
 - $cla>cld$, ocurrió un tm por disminución en el 25% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Cantidad de accesos por día el indicador Accesos por día (ad) presentó comportamientos tales como:
 - Cuando $ad\geq 5$, existe un excesivo acceso al código fuente por parte de ese determinado miembro del proyecto según el 12% de las instancias simuladas.

- Cuando $ad < 5$, existe un acceso al código fuente adecuado por parte de ese determinado miembro del proyecto según el 88% de las instancias simuladas.
- ✓ Durante la simulación de la métrica Cantidad de Modificaciones al código el indicador Cantidad de Modificaciones por día (cmd) presentó comportamientos tales como:
 - $cmd \geq 3$, existió un excesivo número de modificaciones por parte de un usuario en el código fuente en el 15% de las instancias simuladas.
 - $cmd < 3$, existió un adecuado número de modificaciones por parte de un usuario en el código fuente en el 71% de las instancias simuladas.
 - $cmd = 1$, estuvo en presencia del número más recomendable de modificaciones para un usuario dentro del código fuente en el 14% de las instancias simuladas.

3.2. Proceso de mediciones en entorno real

El proceso de mediciones en entorno real se llevó a cabo durante 45 días hábiles, su fecha de inicio fue el 6 de febrero del 2012 y culminó el 6 de abril del 2012. El objetivo de estas pruebas se basó en el establecimiento de las métricas de manera práctica en cada uno de los procesos críticos de soporte a tiempo real. De esta forma se obtuvieron resultados durante todo este proceso de mediciones en entorno real y en algunos casos se pudo prever algunas posibles salidas de los procesos.

3.2.1. Mediciones en entorno real para el proceso de Monitorización de Servidores

El proceso de Monitorización de Servidores se llevó a cabo durante todas las semanas del período de pruebas establecido. Este período de pruebas duró 9 semanas, llevándose a cabo un promedio de 1,9 veces por semanas.

- ✓ Durante el proceso de mediciones en entorno real la métrica Tiempo de conexión con servidor central manifestó su comportamiento a través del indicador tiempo de conexión (tc) el cual tuvo un valor máximo de tres segundos, es decir $tc = 3$ segundos, siendo la la conexión con el servidor central rápida en el 100% de las pruebas realizadas.
- ✓ Durante el proceso de mediciones en entorno real la métrica Espacio disponible en disco duro manifestó su comportamiento a través de los indicadores Espacio Libre de Disco(eld) y Capacidad de Disco(cd) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $eld \leq 0,15 * cd$, se cumplió solamente 5 de las 17 veces que se llevó a cabo esta prueba para un índice probabilístico de un 29%. Es importante destacar que esta fórmula se

cumplió en intervalos aproximados de 15 días, por lo que se puede arribar a la conclusión que se debe de liberar espacio en al menos uno de los discos duros cada 15 días.

- Para el caso $eld > 0,15 * cd$, se cumplió en un 71%, siendo este un caso favorable para el rendimiento del sistema operativo.
- ✓ Durante el proceso de mediciones en entorno real la métrica Cantidad de procesos zombies manifestó su comportamiento a través del indicador Cantidad de Procesos zombies(cpz) el cual tuvo un valor de 0, es decir $cpz=0$ en el 100% de las pruebas realizadas. De acuerdo a los resultados se puede concluir que el Sistema Operativo instalado en los servidores presenta un buen rendimiento dependiendo del indicador cpz.
- ✓ Durante el proceso de mediciones en entorno real la métrica Porcentaje de uso del procesador manifestó su comportamiento a través del indicador Contador Procesador (cp) el cual se manifestó como se muestra a continuación:
 - Para el caso $cp < 50$, la CPU del servidor presentó un uso reducido favoreciendo un buen funcionamiento de este para un 94% de las pruebas realizadas.
 - Para el caso $50 \leq cp < 80$, la CPU del servidor presentó un uso elevado dependiente del consumo de las aplicaciones que se ejecutan sobre este para un 6% de las pruebas realizadas.
 - El caso $cp > 80$ no ocurrió durante las pruebas, por lo que las CPU de los servidores no presentaron un uso crítico dependiente de los recursos de hardware.
- ✓ Durante el proceso de mediciones en entorno real la métrica Por ciento de consumo de la memoria RAM en cuanto a su capacidad manifestó su comportamiento a través de los indicadores Uso de RAM (ur) y Capacidad de RAM (cr) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $ur \leq cr * 0,7$, la memoria RAM presentó un uso adecuado para el servidor en el 88% de las pruebas realizadas.
 - Para el caso $ur > cr * 0,7$, la memoria RAM presentó un uso deficiente para el servidor en el 12% de las pruebas realizadas.

3.2.2. Mediciones en entorno real para el proceso de Descarga

- ✓ Durante el proceso de mediciones en entorno real la métrica Tiempo de recepción de la descarga manifestó su comportamiento a través de los indicadores Tiempo de recepción (tr), Inicio de Descarga (id), Fecha de Recepcionar Descarga (fr) y Fecha de Solicitar descarga (fs) los cuales se manifestaron como se muestra a continuación:

- Para el caso $tr \leq 24$, el indicador id se consideró Adecuado en el 100% de las pruebas realizadas.
 - Cuando $tr > 24$, el indicador id se considera Deficiente en el 0% de las pruebas realizadas.
- ✓ Durante el proceso de mediciones en entorno real la métrica Rapidez de la descarga manifestó su comportamiento a través de los indicadores Nivel de Rapidez (nr), Fecha de Inicio (fi) y Fecha de cumplimiento (fc). Esta métrica pudo ser validada a través de 61 descargas hechas desde los inicios del curso 2011-2012 hasta la actualidad, la primera de estas tuvo lugar el 7 de septiembre del 2011 y la prueba finalizó con la descarga solicitada el 8 de marzo del 2012. Los datos relacionados a estas descargas se encuentran disponibles en el Redmine. Las estadísticas obtenidas en las pruebas responden a la siguiente fórmula: $nr = fc - fi$ mostrando resultados tales como:
- Para el caso $nr \leq 7$ días, la descarga se clasificó en Descarga Rápida en el 41% de las pruebas realizadas.
 - Para el caso $7 < nr \leq 14$ días, la descarga se clasificó en Descarga Lenta en el 41% de las pruebas realizadas.
 - Para el caso $nr > 14$ días, la descarga se clasificó en Descarga Muy lenta en el 18% de las pruebas realizadas.

3.2.3. Mediciones en entorno real para el proceso de Protección del código fuente. Subproceso de Gestión de las Listas de Control de Acceso (ACL)

Para la realización de de estas mediciones se tomó como referencia a la línea productiva de Soluciones Integrales. La pruebas se realizaron al menos una vez por semana durante el período establecido de 45 días hábiles.

- ✓ Durante el proceso de mediciones en entorno real la métrica Cantidad de permisos de acuerdo a roles manifestó su comportamiento a través de los indicadores Cantidad de Miembros de las ACL (cm) y Matrícula total del proyecto (mp) los cuales se manifestaron como se muestra a continuación:
- Para el caso $cm > 0,5 * mp$, existen demasiadas ACL en el proyecto y esto puede ser un factor crítico para la seguridad del código fuente, esto se evidenció en el 0% de las pruebas realizadas.
 - Para el caso $cm \leq 0,5 * mp$, el número de ACL creadas es limitado siendo este un aspecto favorable para la seguridad del código fuente en el 100% de las pruebas realizadas.

- ✓ Durante el proceso de mediciones en entorno real la métrica Coincidencia en cuanto a cantidad de permisos manifestó su comportamiento a través de los indicadores Cantidad de permisos establecidos (ce) y Cantidad de permisos recepcionados (cr) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $ce \neq cr$, existe descoordinación entre las partes responsables de establecer y recepcionar los permisos para establecer las ACL, este aspecto se manifestó en el 0% de las pruebas realizadas.
 - Para el caso $ce = cr$, existe una adecuada coordinación en la creación de permisos para el establecimiento de las ACL en el 100% de las pruebas realizadas.

- ✓ Durante el proceso de mediciones en entorno real la métrica Coincidencia de permisos recepcionados con respecto a la cantidad de permisos dentro de las ACL manifestó su comportamiento a través de los indicadores Cantidad permisos recepcionados (cr) y Cantidad permisos dentro de la ACL (cd) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $cr \neq cd$, las ACL creadas por el Administrador de Servidores carecen de similitud con respecto a la cantidad de permisos establecidos por el Administrador de Configuración, esto se evidenció en un 11% con respecto a las pruebas realizadas
 - Para el caso $cr = cd$, las ACL creadas por el Administrador de Servidores coinciden en cantidad de permisos con lo establecido por el Administrador de Configuración en el 89% de las pruebas realizadas.

- ✓ Durante el proceso de mediciones en entorno real la métrica Cantidad de ACL distribuidas con respecto a la cantidad de ACL administradas manifestó su comportamiento a través de los indicadores Cantidad ACL distribuidas (cd) y Cantidad ACL administradas (ca) los cuales se comportaron como se muestra a continuación:
 - Para el caso $cd \neq ca$, existe descoordinación entre la cantidad de ACL distribuidas al Administrador de Configuración y las que este administra en realidad, esto no se evidenció en ninguna de las pruebas realizadas.
 - Cuando $cd = ca$, el Administrador de Configuración administra al menos la misma cantidad de ACL que le han sido distribuidas siendo esto un aspecto favorable en el proceso de Gestión de las ACL, esto se evidenció en el 100% de las pruebas.

3.2.4. Mediciones en entorno real para el proceso de Protección del código fuente. Subproceso de Acceso al código

El proceso de Acceso al código encierra tanta importancia como el proceso de Gestión de las ACL. Las pruebas que se llevaron a cabo con las métricas elaboradas para el proceso de Acceso al código se pusieron en práctica en la línea productiva de Soluciones Integrales.

- ✓ Durante el proceso de mediciones en entorno real la métrica Factor de coincidencia de cambios con comentarios manifestó su comportamiento a través de los indicadores Factor de coincidencia (fc), Cantidad de modificaciones (cm) y Cantidad Comentarios (cc) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $fc > 0$, existió disparidad en cuanto a la cantidad de modificaciones llevadas a cabo en el código fuente y los comentarios realizados sobre estas modificaciones, esto se evidenció en el 11% de las pruebas realizadas.
 - Para el caso $fc < 0$, existió un excesivo número de comentarios y por tanto se recomienda simplificarlos en vista que haya al menos un comentario por cada cambio realizado, esto no se evidenció en ninguna de las pruebas realizadas.
 - Para el caso $fc = 0$, la cantidad de comentarios coincidió con la cantidad de modificaciones llevadas a cabo en el código fuente en el 89% de las pruebas realizadas.
- ✓ Durante el proceso de mediciones en entorno real la métrica Coincidencia de fechas manifestó su comportamiento a través de los indicadores Fecha de Modificación (fm) y Fecha de Comentario (fc) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $fm \neq fc$, existió descoordinación entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha en el 22% de las pruebas realizadas.
 - Para el caso $fm = fc$, existió una coordinación adecuada entre los cambios realizados y sus respectivos comentarios en cuanto a la fecha en el 78% de las pruebas realizadas.
- ✓ Durante el proceso de mediciones en entorno real la métrica Tipo de Modificación con respecto a la cantidad de líneas de código manifestó su comportamiento a través de los indicadores Tipo de Modificación (tm), Cantidad líneas de código antes (cla) y Cantidad líneas de código después (cld) los cuales se manifestaron como se muestra a continuación:
 - Para el caso $cla < cld$, ocurrió un tm por agregación en el 21% de las pruebas realizadas.
 - Para el caso $cla = cld$, ocurrió un tm por sustitución en el 73% de las pruebas realizadas.
 - Para el caso $cla > cld$, ocurrió un tm por disminución en el 6% de las pruebas realizadas.

- ✓ Durante el proceso de mediciones en entorno real la métrica Cantidad de accesos por día manifestó su comportamiento a través del indicador Accesos por día (ad) el cual se manifestó como se muestra a continuación:
 - Para el caso $ad \geq 5$, existió un excesivo acceso al código fuente por parte de ese determinado miembro del proyecto en el 11% de las pruebas realizadas.
 - Para el caso $ad < 5$, existió un acceso al código fuente adecuado por parte de ese determinado miembro del proyecto en el 89% de las pruebas realizadas.
- ✓ Durante el proceso de mediciones en entorno real la métrica Cantidad de Modificaciones al código manifestó su comportamiento a través del indicador Cantidad de Modificaciones por día (cmd) el cual se manifestó como se muestra a continuación:
 - Para el caso $cmd \geq 3$, existió un excesivo número de modificaciones por parte de un usuario en el código fuente en el 0% de las pruebas realizadas.
 - Para el caso $cmd < 3$, existió un adecuado número de modificaciones por parte de un usuario en el código fuente en el 22% de las pruebas realizadas.
 - Para el caso $cmd = 1$, se estuvo en presencia del número más recomendable de modificaciones para un usuario dentro del código fuente en el 88% de las pruebas realizadas.

3.3. Interpretaciones generales del proceso de mediciones en entorno real

Debido a la inexistencia de datos históricos en relación a los procesos críticos de soporte del centro DATEC no se pudo establecer una comparación con los resultados obtenidos en las pruebas respecto a años anteriores. Se debe aclarar que los procesos antes mencionados a pesar de tener un comportamiento estable se ponían en práctica de manera desorganizada por el hecho de no estar definidos. Los resultados que se muestran en la presente investigación permitieron medir la eficacia de estos procesos durante su transcurso trayendo consigo una estimación en cuanto a comportamiento bajo circunstancias actuales. Las mediciones en entorno real constituyen el punto de partida para comparaciones posteriores teniendo en cuenta el funcionamiento de los procesos críticos de soporte del centro DATEC, permitiendo de esta manera definir estrategias que conlleven a una mejora continua de estos.

Para establecer una comparación entre los resultados del proceso de simulación y el proceso de pruebas en tiempo real se escogió una métrica por proceso. Las interpretaciones gráficas de estos

resultados brindan estadísticas en cuanto al comportamiento de los indicadores en las métricas que se enuncian a continuación:

- Espacio disponible en disco duro. Proceso de Monitorización de Servidores.
- Rapidez de la Descarga. Proceso de Descarga.
- Cantidad de permisos de acuerdo a roles. Proceso de Protección del código fuente.

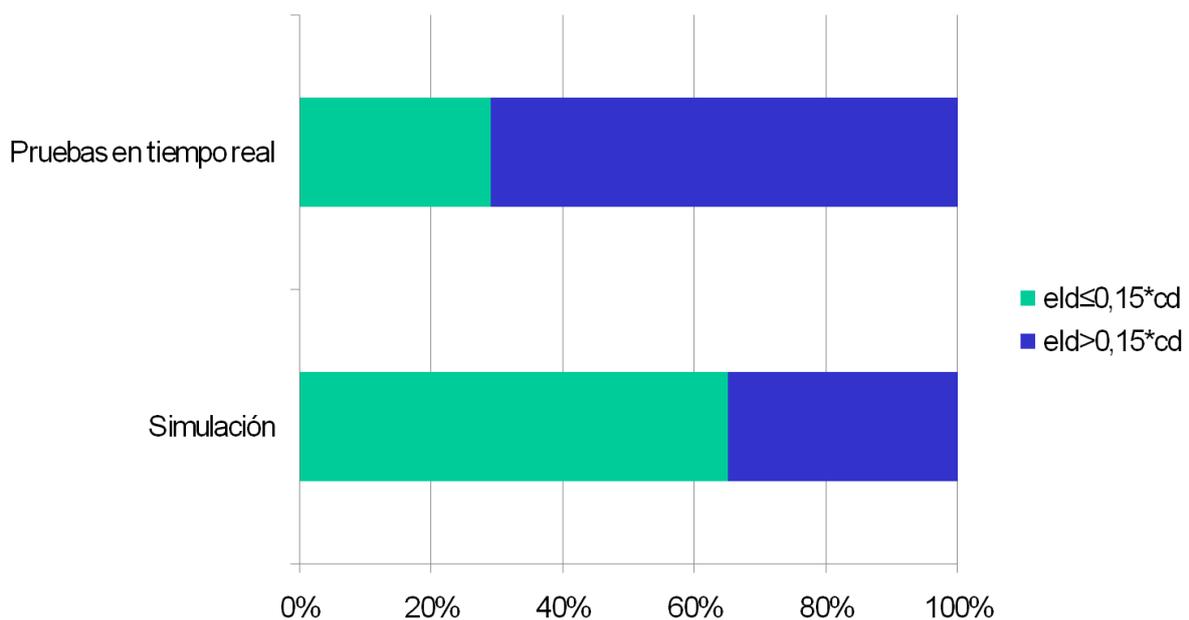


Gráfico 1. Comportamiento de la métrica: Espacio disponible en disco duro. Proceso de Monitorización de Servidores.

El gráfico 1 muestra que el proceso de Monitorización de Servidores en relación a la métrica Espacio disponible en disco duro presentó mejores resultados en las mediciones en entorno real que en las pruebas hechas a través de la simulación. (Ver epígrafes 3.1.2 y 3.2.1)

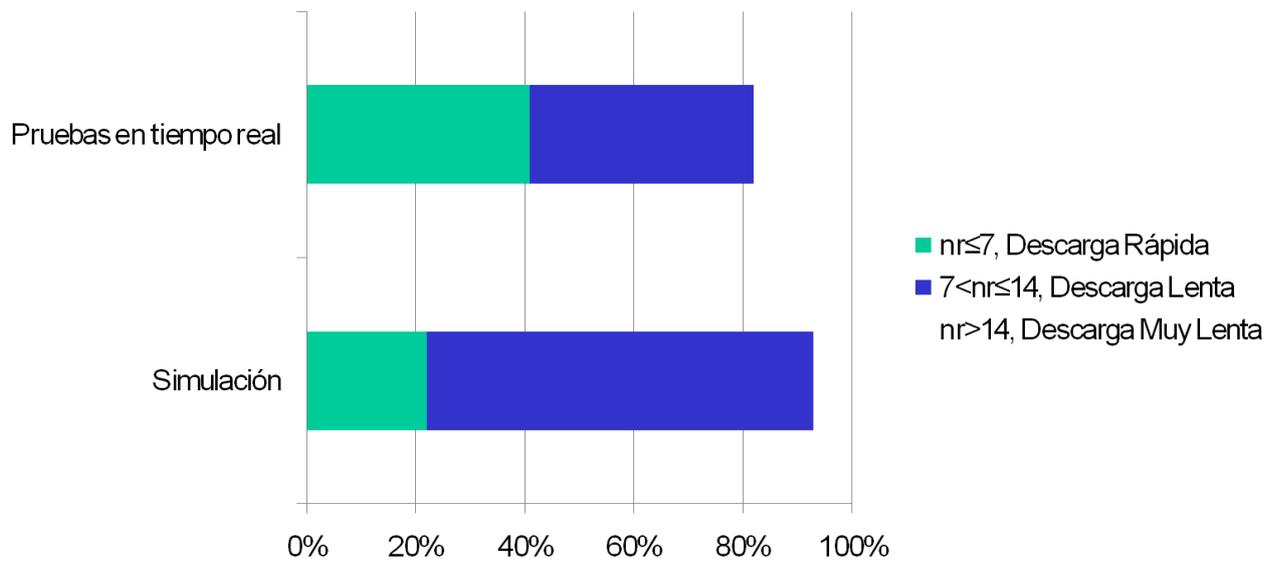


Gráfico 2. Comportamiento de la métrica: Rapidez de la Descarga. Proceso de Descarga

El gráfico 2 muestra que el proceso de Descarga en relación a la métrica Rapidez de la Descarga presentó mejores resultados en las mediciones en entorno real que en las pruebas hechas a través de la simulación. (Ver epígrafes 3.1.4 y 3.2.2)

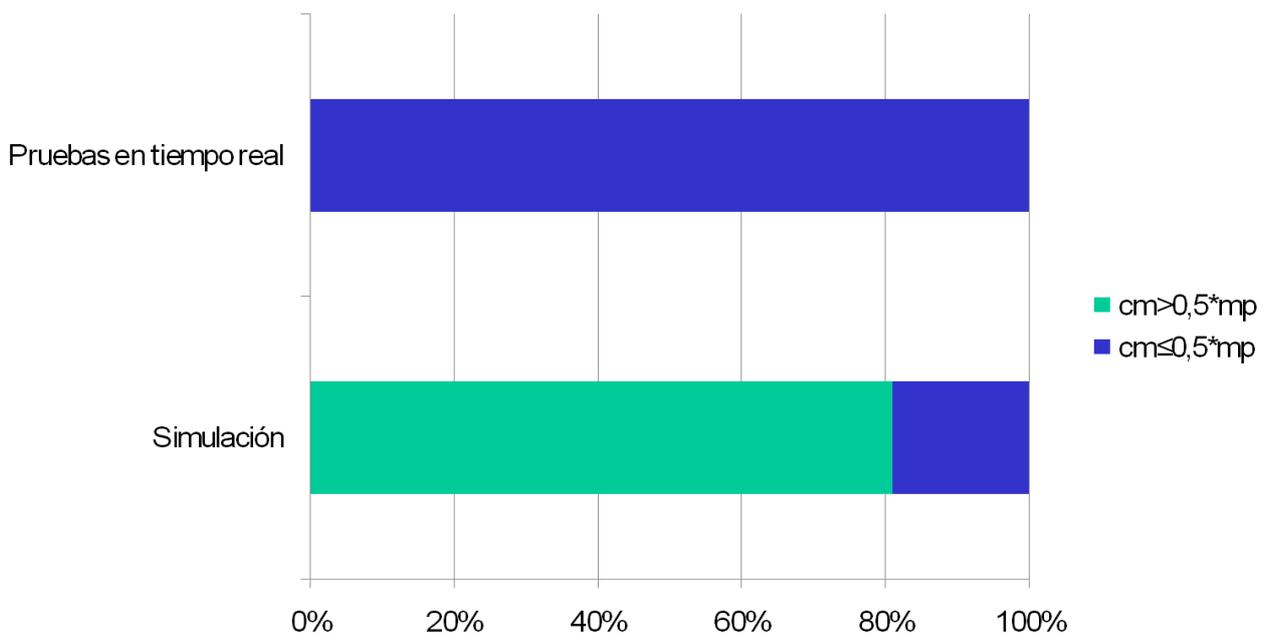


Gráfico 3. Comportamiento de la métrica: Cantidad de permisos de acuerdo a roles. Proceso de Protección del código fuente.

El gráfico 3 muestra que el proceso de Protección del código fuente en relación a la métrica Cantidad de permisos de acuerdo a roles presentó mejores resultados en las mediciones en entorno real que en las pruebas hechas a través de la simulación. (Ver epígrafes 3.1.6 y 3.2.3)

Conclusiones del capítulo

En el presente capítulo se llevó a cabo la aplicación de las métricas propuestas en el capítulo 2 de la presente investigación. Para ello se realizó la simulación de los procesos críticos de soporte del centro DATEC estableciendo valores ficticios en cuanto a cantidad de días, cantidad de horas y cantidad de instancias simuladas. Se llevaron a cabo pruebas en tiempo real durante un período de 45 días hábiles. Los resultados de estas pruebas se brindaron en la unidad de medida *por ciento* (%) estableciéndose de esta manera estimaciones probabilísticas para el futuro.

Conclusiones Generales

- A través de una evaluación del estado del arte se llevó a cabo el análisis de diversos estándares de métricas de calidad seleccionando así la norma ISO 14764 como estándar rector para la elaboración de las métricas de calidad para los procesos críticos del Grupo de soporte del centro DATEC.
- Se diseñaron un total de 21 métricas que brindan diversas interpretaciones de acuerdo a sus resultados. Las métricas elaboradas se establecieron específicamente para las actividades de los procesos seleccionados según lo descrito en la norma ISO 14764.
- Se llevó a cabo la aplicación de la propuesta de métricas a través de la simulación. Se realizaron mediciones en entorno real durante un período de 45 días hábiles en el centro DATEC y los resultados se brindaron en la unidad de medida por ciento (%).

Recomendaciones

- Diseñar métricas de calidad a otros procesos del Grupo de Soporte del centro DATEC tales como: la Gestión de Información de Recursos, la Gestión de Información de Incidencias y las Auditorías internas de Seguridad Informática siguiendo la propuesta del trabajo de diploma.
- Automatizar el sistema de métricas propuesto.

Referencias Bibliográficas

1. **Pressman, Roger.** *Ingeniería de Software*. 1998. 5ta Edición.
2. Diccionario de la Real Academia Española. [En línea] diciembre de 2011. <http://rae.es/rae.html>.
3. Calidad del Software. [En línea] diciembre de 2011. <http://www.slideshare.net/lcahuich/calidad-del-software-presentation>.
4. **Cataldi, Z.1, Lage, F.1, Pessacq, R.2 y García Martínez, R.3,4.** *INGENIERIA DE SOFTWARE EDUCATIVO*. 1996.
5. **Francisco Ruiz, Macario Polo.** [En línea] 2000/2001. <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/mso/trans/s3.pdf>.
6. **Tomalá, Silvia Jazmín Cochea.** [En línea] 2009. <http://www.dspace.espol.edu.ec/bitstream/123456789/4908/1/7708.pdf>.
7. **Lamayzi, Samira.** [En línea] 1998/1999. <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/mso/comple/ISO14764.pdf>.
8. Visual Paradigm. [En línea] <http://www.visual-paradigm.com/>.
9. **Mangas, Alberto Cruz.** [En línea] 2007. <http://bibing.us.es/proyectos/abreproy/4310/fichero/PROYECTO.pdf>.
10. **Madrid, Víctor Javier.** [En línea] 13 de 08 de 2010. <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=bonita>.
11. **Castillo, Paola Alejandra Alvarado.** [En línea] diciembre de 2011. http://fce.unal.edu.co/wiki/images/e/e0/Bonita_Open_Solution..pdf.
12. Real Academia Española. [En línea] http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=proceso.
13. **99-001-2004, Norma Nacional Americana ANSI/PMI.** *Guía de los Fundamentos de la dirección de proyecto (Guía del PMBOOK)*. 2004. 3.

Bibliografía

1. **99-001-2004, Norma Nacional Americana ANSI/PMI.** *Guía de los Fundamentos de la dirección de proyecto (Guía del PMBOOK)*. 2004. 3.
2. *Bonita Open Solution, Open Source BPM Solution (preview)*. Bonita Soft, 2012.
3. Caliad del Software. [En línea] diciembre de 2011. <http://www.slideshare.net/lcahuich/calidad-del-software-presentation>.
4. **Castillo, Paola Alejandra Alvarado.** [En línea] diciembre de 2011. http://fce.unal.edu.co/wiki/images/e/e0/Bonita_Open_Solution..pdf.
5. **Cataldi, Z.1, Lage, F.1, Pessacq, R.2 y García Martínez, R.3,4.** *INGENIERIA DE SOFTWARE EDUCATIVO*. 1996.
6. **Claudia Jiménez Quintana, Lorena Farías Valenzuela, Francisco Pinto.** *Análisis de Modelos de Procesos de Negocios en relación a la dimensión informática*. Santiago de Chile : s.n., 2003.
7. **Elvira Rolón, Francisco Ruiz, Félix García, Mario Piattini.** *APLICACIÓN DE MÉTRICAS SOFTWARE EN LA EVALUACIÓN DE MODELOS DE PROCESOS DE NEGOCIO*. 2005.
8. **Francisco Ruiz, Macario Polo.** [En línea] 2000/2001. <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/mso/trans/s3.pdf>.
9. **Gabriela Noemí Puglla Remache, Lorena del Cisne León Quiñonez.** *Métricas de Proceso y Proyecto*. [En línea] 2012. <http://es.scribd.com/doc/17727554/Metricas-de-Procesos-y-Proyecto>.
10. **Jaime Beltrán Sanz, Miguel A. Carmona Calvo, Remigio Carrasco Pérez, Miguel A. Rivas Zapata, Fernando Tejedor Panchón.** *Guía para una gestión Basada en procesos*. Andalucía : Berekintza, 2003.
11. **Lamayzi, Samira.** [En línea] 1998/1999. <http://alarcos.inf-cr.uclm.es/per/fruiz/cur/mso/comple/ISO14764.pdf>.
12. **Mangas, Alberto Cruz.** [En línea] 2007. <http://bibing.us.es/proyectos/abreproy/4310/fichero/PROYECTO.pdf>.

33. **Madrid, Víctor Javier.** [En línea] 13 de 08 de 2010.
<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=bonita>.
14. **Niurka Martínez Durán, Jorge Luis Cuellar Mondeja.** *Gestión por procesos*. La Habana : s.n., 2011.
15. **Pressman, Roger.** *Ingeniería de Software*. 1998. 5ta Edición.
16. Real Academia Española. [En línea]
http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=proceso.
17. **Sánchez., Ana María García.** *Evaluación de métricas de calidad del software sobre un programa Java*. Madrid : s.n., 2010.
18. **Soft, Bonita.** Bonita Soft. [En línea] 2010. www.bonitasoft.com.
19. **Tomalá, Silvia Jazmín Cochea.** [En línea] 2009.
<http://www.dspace.espol.edu.ec/bitstream/123456789/4908/1/7708.pdf>.
20. *Tutorial Bonita soft parte 1*. 2012.
21. *Tutorial Bonita soft parte 2*. 2012.
22. Visual Paradigm. [En línea] <http://www.visual-paradigm.com/>.
23. **Xavier Franch, Gemma Grau, Carme Quer.** *Un Marco para la Definición de Métricas sobre Modelos de Dependencias entre Actores*. Barcelona, España : 2005.