AUD. 018-2012

INFORME SOBRE LA EVALUACIÓN DE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE DEL IMAS

1. INTRODUCCIÓN

1.1. Origen del Estudio.

El estudio al que se refiere el presente informe, se llevó a cabo de conformidad con el Plan de Trabajo de la Auditoría Interna para el año 2011.

1.2. Objetivo General.

El objetivo general del estudio, consistió en evaluar si la metodología de desarrollo de sistemas del IMAS, guía, controla y documenta satisfactoriamente los procesos de implementación de soluciones basadas en TI.

1.3. Alcance y Periodo de Estudio.

El estudio consistió en evaluar si la metodología considera la definición de requerimientos y análisis de factibilidad; si los controles establecidos en la metodología de desarrollo son suficientes para garantizar razonablemente que se documente el diseño, programación y pruebas de aplicaciones, así como la conversión de datos y puesta en producción; y evaluar el cumplimiento de los controles establecidos en la metodología de desarrollo de sistemas del IMAS. Abarcó el periodo del 1 de abril del 2010 al 30 de junio del 2011, extendiéndose en los siguientes casos:

- En lo referente a la evaluación del proceso de TI relacionado con la metodología de desarrollo de software, se extendió al 19 de diciembre del 2011.
- En relación con la capacitación recibida por los funcionarios al 15 de noviembre, 2011.

Para llevar a cabo este estudio se utilizó como criterios de evaluación la siguiente normativa:

1. Manual de Normas técnicas para la gestión y el control de las Tecnologías de Información, emitido por la Contraloría General de la República.
2. Normas de control interno para el Sector Público.
3. Manual de Políticas de Tecnologías de Información del IMAS, principalmente se utilizó la Política para la Implementación de Soluciones de TI, POL-EDI-20, y la Política para la Gestión de Calidad, POL-EDI-25.

Adicionalmente y para apoyar algunos de los hallazgos determinados, se consideraron los siguientes cuerpos normativos considerados dentro de las mejores prácticas para el manejo de información y gestión de proyectos:

-Objetivos de Control para las Tecnologías de información (COBIT), Versión 4.1

-Guía de los fundamentos para la dirección de proyectos (Guía del PMBOK), del Project Management Institute, cuarta edición.

-Desarrollo Ágil de Software[1] (métodos: Agile, Kanban y Extreme Programming, Scrum), versión 2011

-Metodología de desarrollo de software[2] y enfoques de desarrollo (UML (Lenguaje Unificado de Modelado), POO (Programación Orientado a Objetos), Modelo en cascada, Prototipado, Iterativo, Incremental, Espiral, RAD (Desarrollo Rápido de Aplicaciones)), versión 2011.

El estudio se efectuó de conformidad con el Manual de Normas Generales de Auditoría para el Sector Público (M-2-2006-CO-DFOE), el Manual de procedimientos de Auditoría Interna del IMAS, así como la demás normativa de auditoría interna de aplicación y aceptación general.

1.4. Comunicación verbal de los resultados

En reunión celebrada el día 19 de abril del 2012, se comunicaron los resultados del presente informe a la MSc. Mayra Díaz Méndez, Gerente General, y al Lic. Luis Adolfo González Alguera, Coordinador de Tecnologías de Información, en la cual se efectuaron observaciones que en lo pertinente, una vez valoradas por esta Auditoría Interna, fueron incorporadas en el presente informe.

[1] Desarrollo ágil de software es un marco de trabajo conceptual de la ingeniería de software que promueve iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto. Existen muchos métodos de desarrollo ágil (Agile, Kanban, Extreme Programming y Scrum); la mayoría minimiza riesgos desarrollando software en cortos lapsos de tiempo. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación.

[2] Metodología de desarrollo de software se refiere a un framework que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información. El framework para metodología de desarrollo de software consiste en: una filosofía de desarrollo de programas de computación con el enfoque del proceso de desarrollo de software, y en herramientas, modelos y métodos para asistir al proceso de desarrollo de software Estos frameworks son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología.

2. RESULTADOS

2.1. SOBRE LA METODOLOGÍA DE DESARROLLO DE SOFTWARE DEL IMAS

Se determinó que al 19 de diciembre 2011, no se ha desarrollado y divulgado formalmente una metodología de desarrollo de software para su aplicación en todos los sistemas que gestiona Tecnologías de Información. A pesar de que en el 2003, se inició la confección de un estándar de desarrollo para la construcción de sistemas, éste no fue finalizado y no contempló requerimientos básicos según las buenas prácticas, al enfocarse en la definición de nomenclatura para cada uno de los objetos de la base de datos, omitiendo aspectos claves como la definición de las etapas de desarrollo, como marco de referencia donde se especifique cómo y qué productos hay que obtener durante cada etapa del ciclo de vida de los sistemas, y el o los enfoques o modelos de desarrollo como el UML (Unified Modeling Language), RUP (Rational Unified Process), POO (Programación Orientada a Objetos), RAD (Desarrollo Rápido de Aplicaciones), cascada, prototipado, entre otros.

Al respecto, es importante mencionar lo indicado por el Lic. Carlos Chavarría, Analista Programador de los sistemas sociales, quien señaló lo siguiente:

> "Aunque no tenemos una metodología formal o establecida, **utilizamos una metodología casera** (…).
> **Nosotros tenemos problemas para documentar; analizamos, diseñamos y programamos pero pocas veces documentamos, lo que si hemos intentado es darle seguimiento al diccionario de datos** (…). Es necesario buscar la metodología que mejor se adapte y nos desgaste menos, utilizar herramientas como power designer y demás, pero de momento uno de los problemas que veo es la capacidad para darle seguimiento, esto debido a las cargas de trabajo que manejamos. (…) Uno de nuestros problemas es que no trabajamos bajo algún concepto claro de proyectos (…)."

Adicionalmente, el Analista-Programador Licenciado Marcos Solís, en relación con el desarrollo del Sistema para la Administración de Oficios (SAO), indica que:

> "Cuando yo lo empecé no estaba empapado de cómo lo hacían otros compañeros, por ejemplo los compañeros del SIPO/SABEN usan como nombre de tabla sipm02 para la tabla de personas en sipo;(…) y entonces para modificar el SAO, tendría que hacer un corte y cambiar el sistema pero no es muy viable, cuando yo empecé el sistema no sabía eso, a mi me gustaría que se llamaran las tablas por ejemplo sao_oficioprincipal (…). **Si sería importante documentar porque es ayuda para uno y para los que vienen o lo puedan usar**. (…) **Si ocupamos de fijo una metodología, porque todos trabajamos estandarizados a nosotros mismos, cada uno maneja y documenta el sistema que administra a su forma, porque no hay**

una metodología, uno hace lo básico por si alguien tiene que trabajar en el sistema, como documentar de forma general las tablas, el código, el manual de usuario y entregar mi teléfono."

Sobre el particular, el Manual de Normas técnicas para la gestión y el control de las Tecnologías de Información, emitido por la Contraloría General de la República, indican en lo de interés lo siguiente:

Inciso 1.4.6-Seguridad en **la implementación y mantenimiento de software e infraestructura tecnológica**

La organización debe mantener la integridad de los procesos de implementación y mantenimiento de software (…). Para ello debe: (…) b. Contar con procedimientos claramente definidos para el mantenimiento y puesta en producción del software e infraestructura.

c. Mantener un acceso restringido y los controles necesarios sobre los ambientes de desarrollo, mantenimiento y producción. (…)

Inciso 3.2- Implementación de software

La organización debe implementar el software que satisfaga los requerimientos de sus usuarios y soporte efectivamente sus procesos, para lo cual debe: (…)

b. Desarrollar y aplicar un marco metodológico que guíe los procesos de implementación y considere la definición de requerimientos, los estudios de factibilidad, la elaboración de diseños, la programación y pruebas, el desarrollo de la documentación, la conversión de datos y la puesta en producción, así como también la evaluación post-implantación de la satisfacción de los requerimientos.

c. Establecer los controles y asignar las funciones, responsabilidades y permisos de acceso al personal a cargo de las labores de implementación y mantenimiento de software.

d. Controlar la implementación del software en el ambiente de producción y garantizar la integridad de datos y programas en los procesos de conversión y migración.

e. Definir los criterios para determinar la procedencia de cambios y accesos de emergencia al software y datos, y los procedimientos de autorización, registro, supervisión y evaluación técnica, operativa y administrativa de los resultados de esos cambios y accesos.

f. Controlar las distintas versiones de los programas que se generen como parte de su mantenimiento. (Los subrayados no corresponden al original)

Con respecto a éste tema, las Normas de Control Interno[3], indican lo siguiente:

Inciso 1.4 Responsabilidad del jerarca y los titulares subordinados sobre el SCI

La responsabilidad por el establecimiento, mantenimiento, funcionamiento, perfeccionamiento y evaluación del SCI es inherente al jerarca y a los titulares

[3] Normas de control interno para el Sector Público (N-2-2009-CO-DFOE), aprobadas mediante Resolución del Despacho de la Contraloría General de la República Nº R-CO-9-2009 del 26 de enero, 2009 y publicada en La Gaceta Nº 26 del 6 de febrero, 2009.

subordinados, en el ámbito de sus competencias. /…/ Como parte de ello, deben contemplar, entre otros asuntos, los siguientes: /…/ c. La emisión de instrucciones a fin de que las políticas, normas y procedimientos para el cumplimiento del SCI, estén debidamente documentados, oficializados y actualizados, y sean divulgados y puestos a disposición para su consulta. (…)

Inciso 1.9 Vinculación del SCI con la calidad

El jerarca y los titulares subordinados, según sus competencias, deben promover un compromiso institucional con la calidad y apoyarse en el SCI para propiciar la materialización de ese compromiso en todas las actividades y actuaciones de la organización. A los efectos, deben establecer las políticas y las actividades de control pertinentes para gestionar y verificar la calidad de la gestión, para asegurar su conformidad con las necesidades institucionales, a la luz de los objetivos, y con base en un enfoque de mejoramiento continuo.

Inciso 4.1 Actividades de control

El jerarca y los titulares subordinados, según sus competencias, deben diseñar, adoptar, evaluar y perfeccionar, como parte del SCI, las actividades de control pertinentes, las que comprenden las políticas, los procedimientos y los mecanismos que contribuyen a asegurar razonablemente la operación y el fortalecimiento del SCI y el logro de los objetivos institucionales. Dichas actividades deben ser dinámicas, a fin de introducirles las mejoras que procedan en virtud de los requisitos que deben cumplir para garantizar razonablemente su efectividad. El ámbito de aplicación de tales actividades de control debe estar referido a todos los niveles y funciones de la institución.

Inciso 4.2 Requisitos de las actividades de control

Las actividades de control deben reunir los siguientes requisitos:

e. Documentación. Las actividades de control deben documentarse mediante su incorporación en los manuales de procedimientos, en las descripciones de puestos y procesos, o en documentos de naturaleza similar. Esa documentación debe estar disponible, en forma ordenada conforme a criterios previamente establecidos, para su uso, consulta y evaluación.

f. Divulgación. Las actividades de control deben ser de conocimiento general, y comunicarse a los funcionarios que deben aplicarlas en el desempeño de sus cargos. Dicha comunicación debe darse preferiblemente por escrito, en términos claros y específicos.

Inciso 4.5 Garantía de eficiencia y eficacia de las operaciones

El jerarca y los titulares subordinados, según sus competencias, deben establecer actividades de control que orienten la ejecución eficiente y eficaz de la gestión institucional. Lo anterior, tomando en cuenta, fundamentalmente, el bloque de legalidad, la naturaleza de sus operaciones y los riesgos relevantes a los cuales puedan verse expuestas, así como los requisitos indicados en la norma 4.2.

Asimismo, la Política para la Implementación de Soluciones de TI, POL-EDI-20 del IMAS, en lo de interés señala:

> 20. Debe existir una metodología detallada que sirva como guía para la implementación de toda solución informática.
> 21. La implementación, calidad, monitoreo y mantenimiento de metodología y estándares serán realizados por diferentes personas, buscando obtener una adecuada segregación de funciones.
> 22. La metodología propuesta deberá ser el documento formal a ser utilizado en la implementación de todos los sistemas y aplicaciones realizados por miembros de la Institución o por terceros que se contraten para realizar estas labores
> 23. La metodología será revisada periódicamente por una persona designada por el Coordinador del Área de Desarrollo Informático, y cualquier modificación a este documento deberá ser aprobado por la Subgerencia Administrativa Financiera antes de realizarse. (El subrayado no consta en el original)

Por otra parte, la Política para la Gestión de Calidad, POL-EDI-25 del IMAS, indica lo siguiente:

> 16. La Institución debe adoptar, mantener y documentar estándares de desarrollo que incluyan los aspectos siguientes: pautas de nomenclatura de codificación, formatos de archivos, diseño para esquemas y diccionario de datos, desarrollo para la interfaz de usuario, inter-operabilidad, así como, cualquier otro aspecto que se considere necesario.
> 18. Es responsabilidad del desarrollador determinar los requerimientos de usuario y alinearlos con los estándares y prácticas del Área de Desarrollo Informático. (El subrayado no consta en el original)

Al no estar definida formalmente una metodología de desarrollo de software, según revisión realizada al 19 de diciembre del 2011, se identificó que en el Área de Tecnologías de Información se utilizan prácticas y procedimiento que no son uniformes, pues para un sistema como el de Identificación de población objetivo y atención de beneficiarios (SIPO/SABEN) se utilizan procedimientos un poco más rigurosos y para otro sistema como el Sistema para la administración de oficios (SAO), se utilizan procedimientos menos detallados que son informales y casi no documentan las labores realizadas. De conformidad con la revisión efectuada la aplicación de controles no es uniforme y tampoco es consistente pues no se aplica el mismo control en todos los casos.

A continuación se resume en la siguiente tabla los controles aplicados por Tecnologías de Información según las fases identificadas en dicho proceso, durante el desarrollo y actualización de software:

ETAPAS DE DESARROLLO IDENTIFICADAS EN EL ÁREA DE TI	CONTROLES APLICADOS EN SISTEMA SIPO/SABEN	CONTROLES APLICADOS EN SISTEMA SAO	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SIPO/SABEN	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SAO
Solicitud de requerimiento por usuarios	Se comunican formalmente los requerimientos a TI por medio de una boleta de requerimientos.	El usuario llama por teléfono, comenta, envía correo u oficio a TI u encargado de administrar el	Satisfactorio. Las solicitudes de requerimientos se documentan formalmente a través de una boleta diseñada para registrar el detalle de nuevas solicitudes.	Las solicitudes de requerimientos de los usuarios se realizan por lo general de manera informal, no se utiliza ningún procedimiento

ETAPAS DE DESARROLLO IDENTIFICADAS EN EL ÁREA DE TI	CONTROLES APLICADOS EN SISTEMA SIPO/SABEN	CONTROLES APLICADOS EN SISTEMA SAO	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SIPO/SABEN	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SAO
		sistema sobre el requerimiento	Dicho proceso es consistente.	formal, para que los usuarios soliciten requerimientos. Así, las solicitudes de requerimientos no se documentan de manera consistente.
Análisis Análisis de factibilidad, viabilidad técnica Diagramas	El programador analiza la solicitud comunicada por el LESIIS para evaluar su factibilidad. Consulta diccionario de datos Revisa código fuente	El programador analiza la solicitud para evaluar su factibilidad y comunicar al usuario en caso que no sea realizable.	El análisis realizado es informal, los analistas-programadores se reúnen con el LESIIS y lo conversan, pero no se documenta si es factible técnicamente el requerimiento o no. Es importante destacar lo señalado por el Lic. Carlos Chavarría que a octubre del 2011, no habían recibido requerimientos no factibles, sin embargo no se documenta dicha situación. En el área de Tecnologías de Información no se generan diagramas tipo UML (Lenguaje Unificado de Modelado) u otros según sanas prácticas, por lo que no utilizan diagramas para ejecutar análisis, durante el diseño y desarrollo de aplicaciones. Sólo se consulta el diccionario de datos y se revisa el código fuente. Adicionalmente, no se tiene un diagrama entidad relación, de todo el sistema por la cantidad de tablas de la base de datos SIPAS, ni de los procesos principales.	El análisis realizado por el analista-programador es informal y no se documenta. Cuándo es solicitado un cambio, el analista-programador analiza su factibilidad y realización cuando se programa en la herramienta, no se hace un análisis formal antes de la etapa de diseño y programación. No se realiza ningún tipo de diagrama. No se apoya el análisis en diagramas u otra documentación formal.
Diseño y Programación	-Modifican interfaz en ambiente pruebas.	-Modifican interfaz en ambiente pruebas.	Los analistas-programadores de manera consistente y suficiente modifican la interfaz y el código fuente	El programador de manera consistente y suficiente, modifica la interfaz y el código

ETAPAS DE DESARROLLO IDENTIFICADAS EN EL ÁREA DE TI	CONTROLES APLICADOS EN SISTEMA SIPO/SABEN	CONTROLES APLICADOS EN SISTEMA SAO	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SIPO/SABEN	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SAO
	-Modifican código fuente en ambiente pruebas. -Compilan	-Modifican código fuente en ambiente pruebas. -Compilan	en ambiente de pruebas. Los analistas-programadores no utilizan diagramas como insumo durante el diseño y programación. Si mantienen un diccionario de datos, el cual se encuentra en un 90% de desarrollo, el cual si es consultado. Los programadores utilizan nomenclatura para nombrar objetos de base de datos, es una práctica consistente. No hay estándares para el diseño de la interfaz de usuario y formatos de archivos inter-operabilidad, aunque se sigue un mismo estilo en el diseño de la interfaz.	fuente en ambiente de pruebas. El programador no utiliza, ni realiza diagramas como insumo durante el diseño y programación. En SAO no se utilizan estándares para el diseño de la interfaz de usuario y formatos de archivos inter-operabilidad, aunque se sigue un mismo estilo en el diseño de la interfaz. El analista-programador no utiliza nomenclatura y estándares en la mayoría del código fuente para nombrar objetos de base de datos. Sólo a través de los últimos cambios realizados en el sistema, el técnico ha incluido las nuevas variables siguiendo la nomenclatura usada en SIPO/SABEN. Práctica no consistente.
Documentación	-Se modifica el diccionario de datos -Se documenta el código fuente -Se utiliza parte de un documento denominado "estándar desarrollo para la construcción de sistemas" creado en el 2003 el cual	-Se modifica documento excel -Se documenta código fuente.	-Los analistas-programadores documentan el código fuente de manera general y no estandarizado. Es una práctica no consistente. -El diccionario de datos está en un 90%, falta aproximadamente un 10% para su finalización. -Los programadores aplican estándares para nombrar los	-Como una iniciativa del analista-programador, se ha realizado un documento informal en excel, que indica el nombre del campo, tipo, tamaño y descripción por tabla. No es suficiente, es muy general, ya que se carecen de diagramas entidad-relación, documentación de llaves primarias, foráneas, entre otras.

ETAPAS DE DESARROLLO IDENTIFICADAS EN EL ÁREA DE TI	CONTROLES APLICADOS EN SISTEMA SIPO/SABEN	CONTROLES APLICADOS EN SISTEMA SAO	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SIPO/SABEN	RESULTADO/VARLORACION DE LA APLICACIÓN DEL CONTROL EN SAO
	no fue oficializado.		objetos de la base de datos (tablas, campos, llaves primarias, foráneas, vistas, entre otros.	-Se documenta el código fuente de manera no estandarizada. No es una práctica consistente.
Pruebas	-El programador realiza pruebas -Se verifica con usuario	-El programador realiza pruebas -Se verifica con usuario	-Las pruebas las realizan los analistas-programadores sin un plan formal de pruebas. No obstante, en la boleta de requerimientos, el usuario, jefe de TI y técnico, firman la aceptación del cambio, y se documenta y archiva.	-El proceso de pruebas es informal, ya que se ejecuta cada vez que el técnico realiza algún cambio, pero no se documenta, se realiza sin seguir algún plan formal de pruebas y no se evidencia la aceptación formal del cambio por parte del usuario.
Implementación	-Liberar aplicación en ambiente producción -Instalación -Comunicación de cambios	-Liberar aplicación en ambiente producción -Instalación -Comunicación de cambios	Se comunican formalmente los cambios a través de la boleta de requerimientos. Se actualiza boleta de requerimientos y se archiva como una boleta tramitada por la secretaría de TI. Es una práctica consistente.	-No se comunican los cambios formalmente. -La liberación de la aplicación en ambiente producción y la instalación es suficiente.

Como puede observarse, las prácticas utilizadas cotidianamente por los programadores son insuficientes para documentar y controlar adecuadamente las labores de desarrollo y mantenimiento de sistemas, y presentan brechas importantes con respecto a los controles básicos establecidos en las metodologías de desarrollo consideradas dentro de las sanas prácticas. Es por lo anterior, que en definitiva se requiere elaborar e implantar una metodología que permita controlar adecuadamente las labores de desarrollo, documentar las tareas efectuadas, y garantizar su aplicación consistente.

Al no documentarse y estandarizarse formalmente las etapas de desarrollo y los productos que se deben generar en cada etapa de desarrollo, provoca que el alcance y forma de aplicar controles varíe dentro del mismo sistema y entre un sistema y otro.

Por tanto, un control, actualmente puede omitirse, aplicarse de manera diferente en el mismo sistema, o no documentarse por lo que limita su evaluación y seguimiento, o aplicarse de manera consistente. Por ejemplo, según la información descrita en el cuadro anterior, el análisis de requerimientos, la utilización de nomenclatura y comunicación es consistente en SIPO/SABEN pero no en SAO; asimismo el documentar de manera no estandarizada el código fuente y el no realizar ni utilizar como herramientas de análisis los diagramas es una debilidad en el desarrollo y mantenimiento de ambos sistemas. Finalmente en SIPO/SABEN se ejecutan pruebas pero sin que se evidencie el plan formal de pruebas, y en SAO no se documenta ningún control de la etapa de pruebas, aunque informalmente si las ejecute el

analista-programador. Los anteriores ejemplos ilustran la falta de uniformidad y consistencia en la aplicación de las prácticas de control relacionadas con el desarrollo, causadas esencialmente por la ausencia de una metodología formalmente establecida.

Es importante, reconocer el esfuerzo de los analistas-programadores por documentar, a pesar de la inexistencia de una metodología de desarrollo de software. No obstante, a pesar que se aplican las mismas etapas de desarrollo para todos los sistemas, los controles utilizados no se aplican de manera estandarizada y consistente entre sistemas y no son suficientes para garantizar una documentación razonable durante el desarrollo de software. Esta situación aumenta el riesgo de mantenibilidad de sistemas ante la eventual rotación de personal, y prolonga la duración de la curva de aprendizaje para nuevos programadores que desconocen la forma en que los sistemas están diseñados internamente.

Ante la ausencia de prácticas y procedimientos formalmente establecidos, los programadores recurren a su propia iniciativa para documentar y diseñar programas, utilizando técnicas que eventualmente no son las más apropiadas y que contengan controles efectivos y suficientes para gestionar adecuadamente este proceso. Es así como el objetivo de estructurar, planear y controlar el proceso de desarrollo en sistemas de información para garantizar soluciones acordes con los requerimientos y necesidades del usuario puede verse afectado

Al respecto, es importante mencionar lo indicado por el Lic. Carlos Chavarría, Analista Programador de los sistemas sociales, quien señaló que:

> "Nosotros en alguna medida somos especializados en diversos módulos de las aplicaciones, transferencias o pagos es Barberena, si es algo de SIPO es más conmigo, etc., si alguien falla en ese engranaje podemos tener problemas serios y por desgracia esa especialización que menciono es inevitable porque todos no podemos asumir todo. (…) No es sólo conocer el lenguaje de programación y la base de datos sino el entorno Institucional que es una fortaleza que tenemos los que estamos acá y que hemos ido obteniendo con el tiempo, y sí, esa dependencia de personas es algo que hemos visto pero es difícil solucionarlo."

Así, dada la complejidad en algunos sistemas como el SIPO/SABEN, es importante evitar formalizar una metodología inflexible, difícil de aplicar en todos los casos, que provoque que los técnicos encargados se salten pasos o lineamientos definidos en dicha metodología; y debe permitir un alto grado de reacción y flexibilidad para responder a los requerimientos cambiantes del negocio, sujeta a una mejora continua y garantizando que se produzca documentación que haga eficiente la producción y el mantenimiento de sistemas.

2.2. SOBRE LA CAPACITACIÓN DE LOS FUNCIONARIOS ENCARGADOS DE DESARROLLAR Y ACTUALIZAR SOFTWARE DE TI

A pesar que los analistas-programadores de Tecnologías de Información han recibido capacitación en herramientas de programación y bases de datos como Power Builder, Visual Basic, SQL, ASP, entre otras, se determinó, que dichos analistas no han recibido capacitaciones relacionadas con metodologías de desarrollo de software enfocadas por ejemplo en estándares como el UML para la diagramación,

documentación, entre otros y en administración de proyectos. Adicionalmente, sólo el 50% de los programadores han recibido capacitación en la herramienta Power Designer[4], pero enfocado en la utilización de la herramienta y no en técnicas para diagramar y documentar.

En relación con lo anterior, el Lic. Carlos Chavarría, indicó que "sería importantísimo" realizar capacitaciones de actualización enfocados en la documentación, diagramas, entre otros relacionados con metodologías de desarrollo de software. Adicionalmente, señaló que "Uno de nuestros problemas es que no trabajamos bajo algún concepto claro de proyectos, (…), requerimos capacitación, porque en la parte de administración de proyectos estamos un poco débiles, también nos ayudaría mucho para la metodología de software el trabajar de esa forma, con proyectos.

Asimismo, el analista-programador de Tecnologías de Información, Marco Solís, en relación con la necesidad de capacitaciones para diagramar, indicó lo siguiente: "cuando uno estudió llevó materias relacionadas, pero si uno salió de la universidad por ejemplo hace cinco años y nunca lo puso en práctica porque el día a día te obliga a ir sacando requerimientos como en el SIPO y SABEN, sería importante (…) que nos dieran una capacitación como actualización"

Con respecto a éste tema, las Normas de Control Interno[5], indican lo siguiente:

> 2.4 Idoneidad del personal: El personal debe reunir las competencias y valores requeridos, (…), para el desempeño de los puestos y la operación de las actividades de control respectivas. Con ese propósito, las políticas y actividades de (…) capacitación y otras relacionadas con la gestión de recursos humanos, deben dirigirse técnica y profesionalmente con miras a la contratación, la retención y la actualización de personal idóneo (…) para el logro de los objetivos Institucionales.

Sobre el particular, el conjunto de mejores prácticas para el manejo de información, COBIT[6], en lo de interés señala:

> PO7.4 Entrenamiento del Personal de TI
> Proporcionar a los empleados de TI la orientación necesaria al momento de la contratación y entrenamiento continuo para conservar su conocimiento, aptitudes, habilidades, controles internos y conciencia sobre la seguridad, al nivel requerido para alcanzar las metas organizacionales.
>
> AI 4.4 Transferencia de Conocimiento al Personal de Operaciones y Soporte
> Transferir el conocimiento y las habilidades para permitir al personal de soporte técnico y de operaciones que entregue, apoyen y mantenga la aplicación y la infraestructura asociada de manera efectiva y eficiente de acuerdo a los niveles de servicio requeridos. La transferencia del conocimiento debe incluir al entrenamiento inicial y continuo, el

[4] Herramienta utilizada para construir, diseñar y modelar datos y generar diagramas.

[5] Normas de control interno para el Sector Público (N-2-2009-CO-DFOE), aprobadas mediante Resolución del Despacho de la Contraloría General de la República Nº R-CO-9-2009 del 26 de enero, 2009 y publicada en La Gaceta Nº 26 del 6 de febrero, 2009.

[6] COBIT (Control Objectives for Information and related Technology) versión 4.1, conjunto de mejores prácticas mundialmente aceptados utilizada para planear, implementar, controlar y evaluar el gobierno sobre TIC; incorporando objetivos de control, directivas de auditoría, medidas de rendimiento y resultados, factores críticos de éxito y modelos de madurez. Permite a las empresas aumentar su valor TIC, reducir los riesgos asociados a proyectos tecnológicos, y mejorar las prácticas de planeación, control y seguridad de las Tecnologías de Información.

desarrollo de las habilidades, los materiales de entrenamiento, los manuales de operación, los manuales de procedimientos y escenarios de atención al usuario.

AI4.1 Facilitar la Operación y el Uso
Desarrollar un plan para identificar y documentar todos los aspectos técnicos, la capacidad de operación y los niveles de servicio requeridos, de manera que todos los interesados puedan tomar la responsabilidad oportunamente por la producción de procedimientos de administración, de usuario y operativos, como resultado de la introducción o actualización de sistemas automatizados o de infraestructura.

La situación comentada, es típica de un proceso de gestión de desarrollo de software incipiente, donde existe la percepción de que la documentación de procesos es necesaria pero se genera ocasionalmente y de manera desigual, dado que no se basan en un enfoque estructural o marco de trabajo, lo que provoca que no se invierta en capacitaciones para los funcionarios, en temas relacionados con metodologías para el desarrollo de software, al no tener claridad de qué documentación o productos se deben generar y su importancia, y cómo se desarrolla o actualizan los sistemas a los que TI da mantenimiento.

3. CONCLUSIONES

De conformidad con los resultados obtenidos en el presente estudio, esta Auditoría concluye lo siguiente:

3.1 En relación con las sanas prácticas relacionadas con metodologías de desarrollo de software, se identificó en la Institución un nivel de madurez inicial, ya que Tecnologías de Información ha reconocido que los problemas en relación con la documentación, estandarización de productos que se generan durante el ciclo de vida de sistemas existen y requieren ser resueltos. Sin embargo, no existe un marco de referencia que indique qué hay que obtener a lo largo del desarrollo y mantenimiento de un proyecto de software y cómo hay que obtener los distintos productos conforme se recorre el ciclo de vida de los sistemas, en su lugar existen prácticas que tienden a ser aplicados de forma individual, provocando que aunque se desarrollen y actualicen sistemas funcionales y que operan actualmente, se omita la documentación necesaria y suficiente que facilite el futuro desarrollo y mantenimiento de los sistema y evite riesgos como la dependencia de personal, pérdida del conocimiento, el no hacer eficiente la producción y mantenimiento, al no producirse la documentación necesaria que permita altos grados de reacción y flexibilidad para responder a requerimientos cambiantes del negocio, entre otros.

3.2 Dada la inexistencia de una metodología de desarrollo de sistemas formalmente establecida que guíe, controle y documente satisfactoriamente los procesos de implementación de soluciones basadas en tecnologías, existe un importante riesgo de que se obtengan productos diferentes a los requeridos por el usuario, se utilicen los recursos en forma poco eficiente o se

dependa de las personas que tienen el conocimiento del desarrollo para resolver los problemas, en virtud de la inexistencia de adecuada documentación. Esta deficiencia requiere ser corregida con medidas tendientes a administrar los riesgos comentados y reducir la posibilidad de que los sistemas no sean mantenibles en el tiempo, máxime por el alto grado de dependencia en los sistemas informáticos, que tienen los procesos críticos del IMAS.

3.3 A pesar que existe un alto grado de confianza en el conocimiento de los programadores, dado el desarrollo de sistemas complejos, claves, que operan actualmente y contribuyen significativamente para alcanzar los objetivos del IMAS, se identificó la necesidad de capacitaciones de actualización de conocimientos del personal, enfocadas en metodologías de desarrollo de software, como por ejemplo en estándares como el UML (utilizado para el análisis, implementación y documentación de sistemas orientados a objetos) y administración de proyectos de desarrollo de software y soluciones basadas en tecnologías de información.

4. RECOMENDACIONES

DISPOSICIONES LEGALES SOBRE RECOMENDACIONES

Esta Auditoría Interna respetuosamente se permite recordar a la Máster Mayra Díaz Méndez, en su calidad de Gerente General, y al Lic. Luis Adolfo González Alguera, Coordinador de Tecnologías de Información, que de conformidad con lo preceptuado por el artículo 36 de la Ley General de Control Interno Nº 8292, disponen de diez días hábiles para ordenar la implantación de las recomendaciones, contados a partir de la fecha de recibido de este informe.

Al respecto, se estima conveniente transcribir a continuación, en lo de interés, lo que disponen los artículos 36, 38 y 39 de la Ley Nº 8292:

> Artículo 36._ **Informes dirigidos a los titulares subordinados**. Cuando los informes de auditoría contengan recomendaciones dirigidas a los titulares subordinados, se procederá de la siguiente manera:
>
> a) El titular subordinado, en un plazo improrrogable de diez días hábiles contados a partir de la fecha de recibido el informe, ordenará la implantación de las recomendaciones. Si discrepa de ellas, en el transcurso de dicho plazo elevará el informe de auditoría al jerarca, con copia a la auditoría interna, expondrá por escrito las razones por las cuales objeta las recomendaciones del informe y propondrá soluciones alternas para los hallazgos detectados. /b) Con vista de lo anterior, el jerarca deberá resolver, en el plazo de veinte días hábiles contados a partir de la fecha de recibo de la documentación remitida por el titular subordinado; además deberá ordenar la implantación de recomendaciones de la auditoría interna, las soluciones

alternas propuestas por el titular subordinado o las de su propia iniciativa, debidamente fundamentadas. Dentro de los primeros diez días de ese lapso, el auditor interno podrá apersonarse, de oficio, ante el jerarca, para pronunciarse sobre las objeciones o soluciones alternas propuestas. Las soluciones que el jerarca ordene implantar y que sean distintas de las propuestas por la auditoría interna, estarán sujetas, en lo conducente, a lo dispuesto en los artículos siguientes. /c) El acto en firme será dado a conocer a la auditoría interna y al titular subordinado correspondiente, para el trámite que proceda.

Artículo 38._ **Planteamientos de conflictos ante la Contraloría General de la República**. Firme la resolución del jerarca que ordene soluciones distintas de las recomendadas por la auditoría interna, esta tendrá un plazo de quince días hábiles, contados a partir de su comunicación, para exponerle por escrito los motivos de su inconformidad con lo resuelto y para indicarle que el asunto en conflicto debe remitirse a la Contraloría General de la República, dentro de los ocho días hábiles siguientes, salvo que el jerarca se allane a las razones de inconformidad indicadas. / La Contraloría General de la República dirimirá el conflicto en última instancia, a solicitud del jerarca, de la auditoría interna o de ambos, en un plazo de treinta días hábiles, una vez completado el expediente que se formará al efecto. El hecho de no ejecutar injustificadamente lo resuelto en firme por el órgano contralor, dará lugar a la aplicación de las sanciones previstas en el capítulo V de la Ley Orgánica de la Contraloría General de la República, Nº 7428, de 7 de setiembre de 1994.

Artículo 39._ **Causales de responsabilidad administrativa.** El jerarca y los titulares subordinados incurrirán en responsabilidad administrativa y civil, cuando corresponda, si incumplen injustificadamente los deberes asignados en esta Ley, sin perjuicio de otras causales previstas en el régimen aplicable a la respectiva relación de servicios.

A LA GERENTE GENERAL

4.1. Girar las instrucciones que correspondan y disponer las medidas que estime pertinentes con el propósito de que la Metodología de Desarrollo de Software, que se detalla en la recomendación 4.2 del presente informe, sea aprobada formalmente e implementada, estableciendo un plazo razonable de cumplimiento.

AL COORDINADOR DE TECNOLOGÍAS DE INFORMACIÓN

4.2. Desarrollar, implementar y comunicar formalmente una metodología para el desarrollo estandarizado de software, que indique las etapas y los productos que hay que obtener para su aplicación, durante el ciclo de vida de todos los sistemas que se desarrollen o actualicen en Tecnologías de Información, y que al menos considere lo siguiente:

a. Etapas de desarrollo
b. Enfoques de desarrollo (considerando diagramas)
c. Estándares de codificación, normas de nomenclatura
d. Estándares para la interfaz de usuario
e. Estándares de diseño para esquemas y diccionario de datos
f. Formatos de archivos, inter-operabilidad
g. Actualización de la documentación generada
h. Evaluación del aseguramiento de la calidad sobre el cumplimiento de la metodología de desarrollo de software.

(Ver punto 2.1 del acápite de resultados)

4.3. En coordinación con Desarrollo Humano gestionar la capacitación de los analistas y programadores de software, en temas relacionados con metodologías de desarrollo de software y administración de proyectos, según el nivel requerido por Tecnologías de Información. (Ver punto 2.2 del acápite de resultados)

Hecho por	**Revisado y aprobado**
MATI. Karen Núñez Solano	**MATI. Wady Solano Siles**
PROFESIONAL EJECUTORA	**COORDINADOR AUDITORIA**

AUDITORIA INTERNA
MAYO, 2012