



*Presidencia de la República*

OFICINA PRESIDENCIAL DE TECNOLOGÍAS  
DE LA INFORMACIÓN Y COMUNICACIÓN



# NORTIC A6 2016



NORMA SOBRE EL DESARROLLO Y GESTIÓN  
DEL SOFTWARE EN EL ESTADO DOMINICANO  
SOFTWARE PÚBLICO • SOFTWARE LIBRE • SOFTWARE PROPIETARIO

Santo Domingo, República Dominicana  
Abril 2016





*Presidencia de la República*

OFICINA PRESIDENCIAL DE TECNOLOGÍAS  
DE LA INFORMACIÓN Y COMUNICACIÓN

# NORTIC A6 2016

NORMA SOBRE EL DESARROLLO Y GESTIÓN DEL  
SOFTWARE EN EL ESTADO DOMINICANO

Santo Domingo, República Dominicana

Abril , 2016

---

**NORTIC A6:2016**

**Norma sobre el Desarrollo y Gestión del Software en el  
Estado Dominicano**

Edición: 1era

Oficina Presidencial de Tecnologías de la Información y Comunicación (OPTIC)  
Departamento de Estandarización, Normativas y Auditoría Técnica

Fecha de aprobación: 22 de Enero de 2015

Fecha de lanzamiento: 15 de Abril de 2016

Categoría: A

Serie de documento: 6

Año de publicación: 2016

Versión 0.1.0

Diagramado y Diseñado por el Departamento de Multimedia, OPTIC  
Impreso en República Dominicana





## CONTENIDO

PRÓLOGO.....	vii
MARCO LEGAL.....	xi
INTRODUCCIÓN.....	xvii

### CAPÍTULO I.

#### **Norma sobre el Desarrollo y Gestión del Software en el Estado Dominicano..... 19**

SECCIÓN 1.01 Alcance.....	19
SECCIÓN 1.02 Referencias normativas.....	20
SECCIÓN 1.03 Términos y definiciones.....	20
SECCIÓN 1.04 Licenciamiento.....	21
Sub-sección 1.04.1 Licencias de software propietario.....	21
Sub-sección 1.04.2 Licencias de software libre.....	22

### CAPÍTULO II.

#### **Administración y Desarrollo del Software.....25**

SECCIÓN 2.01 Administración del software.....	25
Sub-sección 2.01.1 Instalación, reinstalación y eliminación del software.....	25
Sub-sección 2.01.2 Actualización del software.....	28
Sub-sección 2.01.3 Adquisición del software.....	30
Sub-sección 2.01.4 Políticas de uso.....	31
Sub-sección 2.01.5 Políticas de administración del software.....	32
Sub-sección 2.01.6 Manejo de errores e incidentes del software.....	33
SECCIÓN 2.02 Desarrollo del software gubernamental.....	35
Sub-sección 2.02.1 Usabilidad.....	35
Sub-sección 2.02.2 Accesibilidad.....	38
Sub-sección 2.02.3 Interoperabilidad.....	39



Sub-sección 2.02.4 Metodología para el desarrollo del software.....	48
SECCIÓN 2.03 Documentación del software.....	53
Sub-sección 2.03.1 Documentación general del software.....	53
Sub-sección 2.03.2 Documentación del código.....	54
Sub-sección 2.03.3 Mantenimiento del software.....	55
SECCIÓN 2.04 Refactorización del código.....	55
SECCIÓN 2.05 Marco de desarrollo recomendado.....	56

### CAPÍTULO III.

<b>Migración del Software.....</b>	<b>59</b>
SECCIÓN 3.01 Directrices generales.....	59
SECCIÓN 3.02 Planificación de la migración.....	59
SECCIÓN 3.03 Aspectos técnicos durante la migración.....	61
Sub-sección 3.03.1 Prueba de la migración.....	62
SECCIÓN 3.04 Inventario general de TIC.....	63

### CAPÍTULO IV.

<b>Gestión e Implementación de Base de Datos.....</b>	<b>65</b>
SECCIÓN 4.01 Implementación de base de datos.....	65
Sub-sección 4.01.1 Implementación de un sistema de gestión de base de datos.....	65
Sub-sección 4.01.2 Diseño y normalización de base de datos.....	68
Sub-sección 4.01.3 Instalación de un sistema de gestión de base de datos.....	69
Sub-sección 4.01.4 Estándares y procedimientos para la implementación del DBMS.....	70
Sub-sección 4.01.5 Uso de base de datos no relacional.....	72
SECCIÓN 4.02 Gestión de base datos.....	73
Sub-sección 4.02.1 Respaldo, recuperación y seguridad de una base de datos.....	74
Sub-sección 4.02.2 Monitoreo y afinamiento de una base datos.....	75



**CAPÍTULO V.**

<b>Repositorio de Software del Estado Dominicano .....</b>	<b>77</b>
SECCIÓN 5.01 Publicación en el repositorio.....	77
SECCIÓN 5.02 Políticas de uso del repositorio.....	78
SECCIÓN 5.03 Software de control de versión.....	79
SECCIÓN 5.04 Recomendaciones sobre el uso del repositorio de software del Estado Dominicano.....	80
GLOSARIO DE TÉRMINOS.....	81
ABREVIATURAS Y ACRÓNIMOS.....	89
BIBLIOGRAFÍA.....	93
ANEXOS.....	95
EQUIPO DE TRABAJO.....	106



## PRÓLOGO

La Oficina Presidencial de Tecnologías de la Información y Comunicación (OPTIC), es el organismo del Estado Dominicano responsable de fomentar el uso de las Tecnologías de la Información y Comunicación (TIC), creado mediante el decreto No. 1090-04, en fecha 3 de septiembre de 2004, como dependencia directa del Poder Ejecutivo, con autonomía financiera, estructural y funcional, a fin de garantizar eficiencia, transparencia, servicios en línea y mecanismos para rendición de cuentas disponibles a favor de la ciudadanía.

Para el aseguramiento del correcto uso e implementación de las TIC en el Estado, la OPTIC elabora y establece las normas y estándares tecnológicos que impulsen el gobierno electrónico en el país.

Estas normas sobre TIC, denominadas NORTIC, son creadas desde el año 2013 por el departamento de Estandarización, Normativas y Auditoría Técnica, bajo el mandato del Ing. Armando García, director general de la OPTIC, y en el gobierno del Presidente de la República Dominicana, Lic. Danilo Medina.

Las NORTIC fueron concebidas para normalizar, estandarizar y tener una herramienta de auditoría para el efectivo uso e implementación de las TIC en la administración pública, con el fin de llegar a la completa homogeneidad y mejora de los procesos entre los organismos gubernamentales.

En este contexto, se han definido 5 categorías o tipos de NORTIC, según el alcance de estas, para ser difundidas e implementadas en toda la administración pública, como se presenta a continuación:

1. **Categoría A**(normas universales), para los aspectos normativos que aplican a todos los organismos gubernamentales.



2. **Categoría B** (normas para los departamentos de TIC), para aquellas normas necesarias y exclusivas a la efectiva gestión de los departamentos o áreas de Tecnologías de la Información y Comunicación (TIC) dentro de los distintos organismos del Estado Dominicano.
3. **Categoría C** (normas municipales), para las normas que aplican a las iniciativas de TIC en los ayuntamientos o municipios.
4. **Categoría D** (normas para embajadas), para las normas que aplican únicamente a las iniciativas de TIC de las embajadas, consulados o misiones en el extranjero.
5. **Categoría E** (normas especiales), para las normas que aplican a organismos gubernamentales con características específicas dependiendo de sus funciones y estructura orgánica, así como para iniciativas, proyectos o programas de Gobierno, en el cual se haga uso de las TIC.

De modo, que esta Norma sobre el Desarrollo y Gestión del Software en el Estado Dominicano, por tener un alcance universal, pertenece a la categoría A; mientras que por ser la sexta NORTIC elaborada en esta categoría, su denominación sería NORTIC A6:2016, siendo los últimos 4 dígitos los referidos al año de lanzamiento de esta norma.

En algunos casos, esta normativa puede presentarse de la forma siguiente: NORTIC A6-1:2016, seguida de trece caracteres (#####-##-#####), donde el número “1” que aparece después del guion (-) especifica la serie del documento (1 para directrices, 2 para guías de implementación, 3 para código de buenas prácticas, entre otros) y los demás caracteres, el Número de Identificación Único (NIU) para cada organismo del Estado.

La evaluación de cada NORTIC es realizada por dos comités, la primera evaluación es ejecutada por el Comité Interno para Evaluación de las Normas (CIEN), el cual está conformado por expertos en TIC dentro de la OPTIC, mientras que la segunda evaluación es realizada por el Comité de Estándares de Tecnologías de la Información y Comunicación (COETIC), el cual está conformado por los responsables de TIC de cada organismo gubernamental, o a quienes la máxima autoridad de cada organismo designe.



En vista de la responsabilidad de la OPTIC en la elaboración de políticas, estrategias y controles de TIC y de los avances en el uso de las tecnologías, de los cuales los organismos gubernamentales no quedan al margen, surge esta normativa con las directrices para garantizar el desarrollo efectivo de las plataformas y los procesos tecnológicos que son implementados por cada uno de dichos organismos.





## MARCO LEGAL

La OPTIC, en su rol de entidad normalizadora sobre el uso e implementación de TIC en la administración pública, ha establecido las directrices por las cuales debe regirse todo organismo gubernamental del Estado Dominicano, tanto para aquellos que están físicamente dentro del país, como para los organismos que se encuentran fuera, como son las embajadas, consulados y misiones en el extranjero.

El marco legal que soporta esta norma está compuesto por las leyes y decretos presidenciales presentados a continuación:

1. El **Decreto 1090-04**, a través del cual se constituye la OPTIC como dependencia directa del poder ejecutivo, donde se establece lo siguiente:
  - **Artículo 3.-** Serán funciones de la Oficina Presidencial de Tecnologías de la Información y Comunicación, diseñar, proponer y definir las políticas, establecer los lineamientos y elaborar las estrategias, planes de acción y programas para promover, desarrollar y garantizar mayor acceso, conectividad e implantación de nuevas tecnologías de la información y comunicación, con el fin de fomentar su uso, como soporte del crecimiento económico y competitividad hacia la sociedad de la información, gestionando los proyectos conforme a tales directrices; y que garanticen el acceso equitativo a los mercados y al sector productivo como esfuerzo para la política de generación de empleo, para mejorar la calidad de vida, crear oportunidades de educación, trabajo, justicia, cultura y recreación, y otros.



- **Artículo 5.-** La Oficina Presidencial de Tecnologías de la Información y Comunicación será responsable de la formulación de políticas y la implementación del proceso de desarrollo e innovación tecnológica para la transformación y modernización del Estado hacia la sociedad de la información, promoviendo la integración de nuevas tecnologías, su compatibilidad, interoperabilidad<sup>[1]</sup> y estandarización en materia de TIC.
  - **Artículo 7.-** La Oficina Presidencial de Tecnologías de la Información y Comunicación podrá proponer políticas para difundir y promover la generación de una cultura de TIC en el país.
  - **Artículo 9.-** La Oficina Presidencial de Tecnologías de la Información y Comunicación deberá velar, asistir y supervisar en los aspectos y políticas relativas a la seguridad y privacidad de la información digitalizada y electrónica en el ámbito del sector público.
2. La **Ley 1-12**, sobre estrategia nacional de desarrollo.
- **Artículo 16.** En el diseño y ejecución de los programas, proyectos y actividades en que se concretan las políticas públicas, deberá promoverse el uso de las tecnologías de la información y comunicación como instrumento para mejorar la gestión pública y fomentar una cultura de transparencia y acceso a la información, mediante la eficientización de los procesos de provisión de servicios públicos y la facilitación del acceso a los mismos.
3. La **Ley 42-2000**, sobre Discapacidad en la República Dominicana.
- **Artículo 7.-** El Estado proveerá de la tecnología adecuada a los centros educativos para la capacitación e información de las personas con discapacidad.

[1] Es la capacidad que tiene un sistema de información para intercambiar datos con otros sistemas con la capacidad de procesarlos.



- **Artículo 47.-** El organismo rector abogará para que los medios de comunicación masiva presenten una imagen comprensiva y exacta de las personas con discapacidad. Asimismo, velará porque estos medios cumplan con las normas éticas y de estilo correspondiente.
4. La **Ley 65-00** sobre Derecho de Autor.
    - **Artículo 2.-** El derecho de autor comprende la protección de las obras literarias y artísticas, así como la forma literaria o artística de las obras científicas, incluyendo todas las creaciones del espíritu en los campos indicados, cualquiera que sea el modo o forma de expresión, divulgación, reproducción o comunicación, o el género, mérito o destino, incluyendo pero no limitadas a:
      - Los programas de computadoras, en los mismos términos que las obras literarias, sean programas fuente o programas objeto, o por cualquier otra forma de expresión, incluidos la documentación técnica y los manuales de uso;
      - Las bases o compilaciones de datos u otros materiales, legibles por máquina o en cualquier otra forma, que por la selección o disposición de sus contenidos constituyan creaciones de carácter intelectual, pero no de los datos o materiales en sí mismos y sin perjuicio del derecho de autor existente sobre las obras que puedan ser objeto de la base o compilación;
  5. La **Ley 340-06** sobre Compras y Contrataciones de Bienes, Servicios, Obras y Concesiones, en donde se establecen los principios y normas generales que rigen la contratación pública, relacionada con los bienes, obras, servicios y concesiones del Estado.
  6. La **Ley 53-07** contra Crímenes y Delitos de Alta Tecnología.
    - **Artículo 1.-** Objeto de la Ley. La presente ley tiene por objeto la protección integral de los sistemas que utilicen tecnologías de la información y comunicación,



y su contenido, así como la prevención y sanción de los delitos cometidos contra estos o cualquiera de sus componentes o los cometidos mediante el uso de dichas tecnologías en perjuicio de personas física o morales, en los términos previstos en esta ley. La integridad de los sistemas de información y sus componentes, la información o los datos, que se almacenan o transmiten a través de estos, las transacciones y acuerdos comerciales o de cualquier otra índole que se llevan a cabo por su medio y la confidencialidad de estos, son todos bienes jurídicos protegidos.

7. La **Ley 107-13**, sobre los derechos de las personas en sus relaciones con la administración pública y de procedimiento administrativo, en donde se regulan los derechos y deberes de las personas y sus relaciones con la administración pública y se establecen los principios que sirven de sustento a esa relación, indicando los procedimientos administrativos.
- **Artículo 4.** Derecho a la buena administración y derechos de las personas en sus relaciones con la administración pública. Se reconoce el derecho de las personas a una buena administración pública, que se concreta, entre otros, en los siguientes derechos subjetivos de orden administrativo:
    - Derecho a no presentar documentos que ya obren en poder de la administración pública o que versen sobre hechos no controvertidos o no relevantes.
  - **Artículo 27.** Actos de instrucción o investigación. Los actos de instrucción o investigación podrán consistir, entre otros, en los siguientes medios:
    - Párrafo I. Las actuaciones para la obtención y tratamiento de la información necesaria para adoptar una decisión bien informada podrán consistir en cualquier medio, como la cooperación, asistencia e intercambio de información con otras



administraciones competentes, o las consultas a los expertos. En los términos establecidos en la legislación o en convenios internacionales, podrá recabarse la colaboración informativa de otras agencias y administraciones especializadas de otros Estados, o de organismos internacionales, al objeto de adoptar la decisión mejor informada, al servicio de los intereses generales.

8. El **Decreto 229-07**, el cual es el instructivo de aplicación de Gobierno Electrónico, contentivo de las pautas generales para el desarrollo de la Estrategia de Gobierno Electrónico en la República Dominicana.
9. El **Decreto 709-07** sobre las normas y estándares elaboradas por la OPTIC.
  - **Artículo 1.-** Se instruye a toda administración pública del Estado Dominicano a cumplir con las normas y los estándares tecnológicos para: (i) el desarrollo de portales gubernamentales, (ii) conectividad interinstitucional, (iii) interoperabilidad tecnológica, (iv) de seguridad, auditoría e integridad electrónica, (v) digitalización de documentos; así como cualquier otra normativa que sea redactada, aprobada y coordinada por la Oficina Presidencial de Tecnologías de la Información y Comunicación (OPTIC), en materia de Tecnología de la Información y la Comunicación (TIC) y Gobierno Electrónico.
10. El **Decreto 615-07**, que Instruye a la OPTIC a coordinar el procedimiento para la elaboración de los inventarios respecto a los programas incorporados a las computadoras y su licenciamiento.
11. El **Decreto 134-14**. Promulgado a los 9 días del mes de abril del año 2014, sobre el Reglamento de Estrategia Nacional de Desarrollo.
12. La **Resolución 51-2013**, que aprueba los modelos de estructura organizativa permitidos para las unidades de TIC de todos los organismos del sector público.



# INTRODUCCIÓN

La Norma sobre el Desarrollo y Gestión del Software en el Estado Dominicano, es un documento que establece las directrices que debe seguir cada organismo del Gobierno Dominicano para el desarrollo y administración del mismo, con el objetivo de que lo desarrollado por dichos organismos sea un activo que agregue valor a los usuarios<sup>[2]</sup> y a toda la ciudadanía.

Esta norma también conocida como NORTIC A6, indica desde el primer capítulo su alcance, el cual comprende todos los organismos gubernamentales de manera obligatoria. Además, se establecen las directrices que deben cumplir respecto al correcto licenciamiento para el uso de los diferentes tipos de software<sup>[3]</sup>.

El capítulo II sobre la administración y desarrollo del software, establece todas las pautas necesarias para la correcta implementación de una metodología del desarrollo del software y una administración eficiente del mismo, para lograr una homogeneidad en todas las aplicaciones de los organismos.

El siguiente capítulo sobre la migración del software, establece las directrices que permiten llevar un proceso de sustitución de un software a otro, desde su planificación inicial, hasta su realización final. También establece el procedimiento, levantamiento, actualización y control de inventario que todos los organismos deben seguir para la gestión de activos de software que se encuentran bajo la responsabilidad del departamento de TIC.

El capítulo IV, aborda todo lo relacionado a la gestión e implementación de base de datos en el Estado dominicano. En el mismo se contemplan tantos aspectos de diseño, implementación, administración, monitoreo y los sistemas de gestión.

[2] Hace referencia a la persona que consume o manipula un producto, servicio o información.

[3] Se refiere a todos los componentes lógicos o intangibles de un sistema de información, tales como programas, aplicaciones, sistemas operativos, entre otros.



El capítulo final, sobre el repositorio de software del Estado Dominicano, se presenta la metodología de publicación del software en el sitio web oficial del Gobierno Dominicano, [www.softwarepublico.gob.do](http://www.softwarepublico.gob.do), el cual, para evitar la duplicidad de esfuerzo, ponen a disposición de otras organizaciones, las aplicaciones tecnológicas que han sido desarrolladas o adquiridas por los organismos. También este capítulo indica las pautas para establecer un patrón de trabajo mediante la herramienta GIT<sup>[4]</sup>.

[4] Es un sistema de control de versiones de código abierto que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, y estas versiones específicas pueden ser utilizadas más adelante.



# CAPÍTULO I

## NORMA SOBRE EL DESARROLLO Y GESTIÓN DEL SOFTWARE EN EL ESTADO DOMINICANO

La norma indica las directrices que debe seguir cada organismo del Gobierno Dominicano para la gestión e implementación del software público en el Estado, con el objetivo de estandarizar y manejar eficientemente lo relacionado con el trámite por parte de la administración pública. Incluyendo la misma los temas sobre el licenciamiento, alcance, referencias normativas y términos y definiciones, en donde se especifica la metodología que este debe seguir en el Estado Dominicano.

### SECCIÓN 1.01.

### Alcance

Las directrices de la norma deben ser aplicadas por todos los organismos pertenecientes al Poder Ejecutivo, ya sean centralizados, descentralizados, o embajadas, consulados, misiones en el extranjero y municipios.

Entre los organismos centralizados se encuentran los Ministerios y sus dependencias, así como los organismos con nivel de ministerios, viceministerios, organismos adscritos a la Presidencia de la República, consejos y organismos asesores, direcciones generales, oficinas nacionales, procuradurías fiscales, escuelas públicas, hospitales públicos, bibliotecas y museos.

Entre los organismos descentralizados se encuentran las instituciones financieras y las no financieras, organismos reguladores, instituciones de la seguridad social y empresas públicas.



Los organismos pertenecientes al Poder Legislativo, y al Poder Judicial, así como aquellos organismos que entran dentro de la clasificación de “Organismos Especiales” u autónomos, según el Ministerio de Administración Pública (MAP), también pueden implementar los estándares indicados en esta norma como un modelo de buenas prácticas.

## SECCIÓN 1.02.

### Referencias normativas

La NORTIC A6, hace referencia a la norma para la interoperabilidad entre los organismos del Gobierno Dominicano (NORTIC A4), publicada el 10 de junio de 2014, la cual fue tomada como base para la redacción de la sección sobre interoperabilidad. También hace referencia a la norma sobre el uso e implementación de las TIC (NORTIC A1) para los temas sobre licenciamiento e inventario general de TIC.

En la gestión e implementación de bases de datos, se tomó la ISO<sup>[5]</sup>/IEC 9075, para la definición y uso del Lengua de Consulta Estructurado (SQL<sup>[6]</sup>, por sus siglas en inglés).

Para los elementos de datos, formatos intercambiables, intercambio de información y representación de fechas y horas se utiliza la ISO 8601 para establecer un estándar en la presentación de estos particulares.

Para el tema sobre desarrollo de software, se utilizó como referencia la norma ISO 12207, de la Organización Internacional de Normalización (ISO, por sus siglas en inglés), sobre los procesos del ciclo de vida del software y el marco de desarrollo de la metodología Scrum, el cual es un marco para la gestión y desarrollo de software, permitiendo realizar un desarrollo ágil.

## SECCIÓN 1.03.

### Términos y definiciones

Para fines de esta norma, el término “Organismo gubernamental” será utilizado en ciertos casos como “Organismo”.

[5] Es una organización encargada de la creación de normas y estándares internacionales en diferentes áreas como tecnologías, seguridad, servicios, entre otros.

[6] El SQL es un lenguaje de programación utilizado para acceder a bases de datos relacionales.



Los términos “Software”, “Aplicaciones” y “Programas”, para fines de esta guía, se utilizarán indistintamente. Mientras, el término “aplicación web” se refiere a toda aplicación creada con lenguajes y tecnologías estándares de Internet aprobados por la W3C, entre ellos HTML, Javascript y CSS.

Cuando aparezca el término "repositorio de software", será sustituido por la palabra "repositorio".

Cuando en la normativa aparezca el término “Activos”, este se refiere tanto a los activos físicos como a los activos de información.

## SECCIÓN 1.04.

## Licenciamiento

Para el cumplimiento de la Ley 65-00, sobre derecho de autor, los organismos gubernamentales, deben determinar el licenciamiento correcto para todo el software que estos utilicen, sea propietario o libre, evitando de esta manera, la violación de las políticas establecidas para el uso de cada tipo de licenciamiento.

### Sub-sección 1.04.1.

### Licencias de software propietario

- (a) Todo organismo debe aplicar las directrices establecidas a continuación, permitiendo así el cumplimiento de la ley 65-00, sobre derecho autor.
  - (i) Cada software utilizado por los organismos debe incluirse en el inventario general de TIC, siguiendo la periodicidad y las directrices especificadas en la **Sección 3.04 sobre inventario general de TIC**.
  - (ii) Cada software utilizado en los organismos debe contar con la licencia respectiva.
  - (iii) Debe eliminarse aquel software que, requiriendo licencia para su uso, no cuente con la misma.
  - (iv) Solo debe instalarse la cantidad de copias de software permitidas por las licencias que poseen los organismos.
  - (v) No deben reproducirse ni distribuirse copias no autorizadas de software por Internet u otros medios electrónicos.



## Sub-sección 1.04.2.

## Licencias de software libre

- (a) Los organismos que utilicen software libre<sup>[7]</sup> deben cumplir con las siguientes políticas:
  - (i) La licencia de todo software desarrollado por los organismos debe ser abierta.
  - (ii) Debe permitirse la libertad de ejecutar, copiar, distribuir, estudiar el software e incluso modificarlo para cualquier propósito.
  - (iii) Cualquier implementación basada en otro software gubernamental, tomado del repositorio de software del Estado Dominicano, debe tener la misma licencia implementada.
  - (iv) Ningún código fuente<sup>[8]</sup> modificado desde el repositorio del estado, puede ser vendido, ni ofrecido como servicio contratado a los organismos gubernamentales.
- (b) Para la implementación de licencias en el estado, deben estar categorizada por la Open Source Initiative (OSI, por sus siglas en inglés), como populares y ampliamente utilizada, a continuación:
  - (i) En caso de utilizar la Licencia General Pública (GNU<sup>[9]</sup> GPL 3.0, por sus siglas en inglés), el desarrollador debe conservar los derechos de autor, pero permitir su libre distribución, modificación y uso.
  - (ii) Cuando se utilice la Licencia Pública General Reducida de GNU (LGPL, por sus siglas en inglés), debe permitírsele a cualquier persona la copia y distribución de esta licencia.
    - a) Las aplicaciones realizadas bajo esta licencia no deben permitir su modificación.

[7] El software libre es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas.

[8] Es un conjunto de instrucciones redactas en base a las reglas sintácticas de un lenguaje de programación para desarrollar un software determinado.

[9] Es una licencia libre y gratuita con derecho de copia para software y otros tipos de obras.



- (iii) Cuando se utilice la licencia de software Apache 2.0, debe permitírsele al usuario modificar y distribuir las versiones de ese software.
  - a) Las versiones de software realizadas bajo esta licencia deben conservar el derecho de autor.
- (iv) Cuando se utilice la Licencia Pública Mozilla<sup>[10]</sup> (MPL, por sus siglas en inglés), debe permitírsele al usuario ejecutar, copiar, y distribuir el software.
  - a) Debe permitirse la posibilidad de que las nuevas versiones que se realicen del software puedan ser de licencia sea propietaria.
- (v) Cuando se utilice la Licencia Pública Eclipse (EPL<sup>[11]</sup>, por su sigla en inglés), debe permitírsele al usuario utilizar, modificar, copiar y distribuir el software.
  - a) Todas las versiones modificadas bajo esta licencia, no deben estar obligadas a liberar sus propios cambios.
- (vi) Cuando los organismos utilicen la licencia Distribución de Software Berkeley (BSD<sup>[12]</sup>, por sus siglas en inglés), deben garantizar el crédito a los autores del software.
- (vii) Todo organismo que utilice la licencia del Instituto de Tecnología de Massachusetts (MIT, por sus siglas en inglés), deben ser libres para usar el código como deseen y lo puedan modificar.

[10] Cumple completamente con la definición de software de código abierto de la Open Source Initiative (OSI) y con las cuatro libertades del software libre enunciadas por la Free Software Foundation (FSF).

[11] Es una licencia de software de código abierto utilizada por la Fundación Eclipse para su software. Sustituye a la Licencia Pública Común (CPL) y elimina ciertas condiciones relativas a los litigios sobre patentes.

[12] Berkeley Software Distribution o BSD (en español, distribución de software berkeley) fue un sistema operativo derivado del sistema Unix nacido a partir de los aportes realizados a ese sistema por la Universidad de California en Berkeley.



## CAPÍTULO II

### ADMINISTRACIÓN Y DESARROLLO DEL SOFTWARE

En el Estado Dominicano son necesarias directrices y políticas que establezcan el correcto marco de desarrollo y administración del software en la administración pública, por tal razón, en este capítulo se establecen las pautas necesarias para una correcta metodología del desarrollo del software y una administración eficiente del mismo.

#### SECCIÓN 2.01.

#### Administración del software

Los organismos gubernamentales requieren políticas y controles para una correcta administración del software, los cuales permitan un control sobre los mismos para asegurar el correcto funcionamiento y desempeño.

#### Sub-sección 2.01.1.

#### Instalación, reinstalación y eliminación del software

- (a) El organismo gubernamental debe tener políticas para el control de las instalaciones, reinstalaciones o eliminaciones del software, para lo cual se establece lo siguiente:
  - (i) La unidad de administración del servicio TIC (ver **NORTICA1:2014, Directriz 2.01.b.iv.**), debe contar con un personal que asuma la función de las instalaciones, reinstalaciones o eliminaciones del software en los equipos de los usuarios. Para las instalaciones, reinstalaciones o eliminaciones en equipos, tales como servidores<sup>[13]</sup>, base de datos, equipos de redes y comunicaciones entre otros, la unidad de operaciones TIC debe contar con el personal necesario para realizar estas funciones.

[13] Son equipos informáticos que forman parte de una red de datos y que proveen servicio a otros equipos en dicha red, llamados clientes.



- a) El personal designado debe tener conocimientos técnicos de los sistemas operativos y aplicaciones utilizadas en las áreas de operación y administración TIC antes mencionadas.
- (ii) Todo requerimiento o solicitud de instalación, reinstalación o eliminación del software por parte del usuario, debe ser realizado mediante el correo electrónico institucional (ver **NORTIC A1:2014, Subsección 7.04.3. Correo institucional**), mesa de servicio o algún otro medio de comunicación usado por el organismo, detallando como mínimo:
  - Nombre de usuario.
  - Cargo de usuario.
  - Razones de la instalación/reinstalación/eliminación.
  - Programas, aplicaciones o utilitarios a instalar, reinstalar o eliminar.
  - i) Estas solicitudes deben tener un tiempo de respuesta menor a las 24 horas por parte del departamento de TIC, ya sea para establecer un tiempo de solución o para cerrar el requerimiento solucionando el problema.
  - ii) El departamento de TIC debe documentar la solicitud en su banco de conocimiento, en caso de que la misma haya sido a causa de problemas con un software instalado, a fin de tener los procedimientos necesarios para la solución del mismo en casos posteriores.
- (iii) Todo requerimiento o solicitud de instalación, reinstalación o eliminación del software perteneciente a la plataforma de TIC, debe ser aprobado por el personal autorizado del área de Operaciones TIC (ver **NORTIC A1:2014, Directriz 2.01.1.b.iii**).
  - a) En caso de que la solicitud involucre parte de la infraestructura tecnológica sensible, estos cambios





deben realizarse en un ambiente de pruebas<sup>[14]</sup> antes de pasar a un ambiente de producción<sup>[15]</sup>.

- b) En caso de que los requerimientos involucren infraestructuras que administren información de usuarios o del organismo, debe realizarse un respaldo de las informaciones como precaución, en caso de fallas. Ver NORTIC A1:2014, Sub-sección 6.02.4. Respaldo de la información.
- (iv) El departamento de TIC debe comprobar que las solicitudes de instalación, reinstalación o eliminación del software estén sustentadas en licencias auténticas y que las mismas correspondan a su inventario de software.
- (v) Toda solicitud de instalación o reinstalación del software debe cumplir con los requerimientos mínimos de hardware<sup>[16]</sup> especificados por el fabricante como:
- Tipo de procesador.
  - Cantidad de memoria.
  - Cantidad de espacio en disco.
  - Sistema operativo<sup>[17]</sup> o plataforma compatible.
  - Cualquier otro requerimiento especificado por el fabricante.
- (vi) Antes de realizarse la desinstalación de cualquier software de la infraestructura TIC, debe hacerse un respaldo de los archivos o informaciones como precaución, en caso de fallas.
- (vii) Toda instalación, reinstalación o eliminación del software en la plataforma TIC o en los equipos de usuario debe ser documentada.

[14] Es el espacio que se utiliza en el área de TIC para comprobar el funcionamiento de una nueva implementación, con el objetivo de determinar posibles errores y mejoras antes de pasar a producción.

[15] Es donde el personal del organismo realiza sus operaciones diarias e interactúa con el ciudadano. En este ambiente se maneja la información sensible del organismo y cualquier incidente puede afectar la productividad del mismo.

[16] Se refiere a todas las partes físicas o tangibles de un sistema de información.

[17] Es un software utilizado en los sistemas de información para gestionar y administrar los recursos de los dispositivos o computadores.



- a) La documentación debe contener como mínimo:
- Nombre y tipo del software adquirido.
  - Justificación de la instalación, reinstalación o reparación.
  - Equipo donde se realizó la instalación, reinstalación o reparación.
  - Responsable de la instalación, reinstalación o reparación.

### Sub-sección 2.01.2.

### Actualización del software

- (a) El organismo gubernamental debe tener políticas para el control de las actualizaciones de software, para lo cual se establece lo siguiente:
- (i) El departamento de TIC debe confirmar la disponibilidad de los licenciamientos pertinentes antes de cualquier actualización.
- (ii) Para las actualizaciones automáticas de software, deben tomarse en cuenta las siguientes directrices:
- a) Las actualizaciones automáticas deben ser realizadas mediante un servidor de aplicaciones, repositorio o algún otro medio apto para los fines, tomando en cuenta lo siguiente:
- i) Las actualizaciones de software deben ser programadas en horas no laborales o en horas donde se experimente menos uso del ancho de banda de la red.
- ii) El proceso de actualización no debe interrumpir los servicios del organismo.
- iii) El proceso de actualización entre el servidor o repositorio de aplicaciones y el proveedor, no debe afectar el ancho de banda de la red.
- (iii) Toda solicitud de actualización del software debe cumplir con los requerimientos mínimos de hardware especificados por el fabricante como:



- Tipo de procesador.
  - Cantidad de memoria.
  - Cantidad de espacio en disco.
  - Sistema operativo o plataforma compatible.
  - Cualquier otro requerimiento especificado por el fabricante.
- (iv) Todo requerimiento de actualización del software, debe ser realizado mediante el correo electrónico institucional (ver **NORTIC A1:2014, Sub-sección 7.04.3. Correo institucional**), mesa de servicio o algún otro medio de comunicación usado por el organismo, detallando como mínimo:
- Nombre de usuario.
  - Cargo de usuario.
  - Razones de la actualización.
  - Programas, aplicaciones o utilitarios a actualizar.
    - i) Estas solicitudes deben tener un tiempo de respuesta menor a las 24 horas por parte del departamento de TIC, ya sea para establecer un tiempo de respuesta o para la solución del requerimiento.
- (v) Las actualizaciones para estaciones de trabajo deben regirse por las siguientes directrices:
- a) El departamento de TIC debe comprobar que las solicitudes de actualización de software estén sustentadas en licencias auténticas y que las mismas correspondan a su inventario de software.
  - b) Las actualizaciones que involucren sistemas críticos de soporte al usuario deben realizarse en ambientes controlados o simulacros.
    - i) Antes de la realización de actualizaciones, el departamento de Operaciones TIC, debe crear un respaldo de la terminal, en caso de fallas. Ver **NORTIC A1:2014, Sub-sección 6.02.4. Respaldo de la información.**



- (vi) Las actualizaciones para la infraestructura TIC debe registrarse por las siguientes directrices:
  - a) Todas las actualizaciones deben ser evaluadas por el área de Operaciones de TIC. ver **NORTIC A1:2014, Directriz 2.01.1.b.iii.**
  - b) Las actualizaciones que involucren partes críticas de la infraestructura como administrador de correos, administrador de base de datos, sistema operativo o algún software de soporte a áreas gerenciales u operacionales del organismo, deben realizarse en ambientes controlados o simulados.
    - i) Antes de la realización de actualizaciones en ambientes de producción, el departamento de Operaciones TIC, debe crear respaldo de todos los sistemas que puedan verse afectados, en caso de fallas. Ver **NORTIC A1:2014, Sub-sección 6.02.4. Respaldo de la información.**

### Sub-sección 2.01.3.

### Adquisición del Software

- (a) Los organismos gubernamentales que contraten servicios de desarrollo de aplicaciones, deben exigir a los desarrolladores la propiedad exclusiva de la aplicación y código fuente desarrollado.
- (b) Todo software adquirido debe ser libre.
  - (i) Solo debe adquirirse el software propietario en los casos donde el software libre no cumpla con los requerimientos del organismo.
- (c) Cuando el organismo necesite sub-contratar desarrollos de software estos deben ser de proveedores locales.
  - (i) Solo se permite la sub-contratación de desarrollos internacionales cuando los locales no cumplan con los requerimientos del organismo.
- (d) El software de Gestión de Servidores Web (http server y servicios relacionados) debe ser abierta.
- (e) El software de Gestión de Servidores De Voz sobre IP (VOIP, por sus siglas en inglés) debe ser abierta o libre.



- (f) Las soluciones de seguridad en general deben incluir contrato de soporte, particularmente el control de spam, phishing, firewall, antivirus, entre otros.

#### Sub-sección 2.01.4.

#### Políticas de uso

- (a) El departamento de TIC debe seguir las siguientes directrices y políticas para el uso del software:
  - (i) Todo software gubernamental debe ser usado bajo las licencias establecidas por sus autores o por el departamento de TIC, en caso de ser desarrollado por el mismo organismo. Ver **sección 1.04 Licenciamiento**.
  - (ii) Si durante la realización del inventario se encuentra un software no licenciado, tanto en las estaciones de trabajo como en los servidores, el departamento de TIC debe seguir las pautas establecidas en la **directriz 2.01.5.a.i.a.**
  - (iii) Todo software, desarrollado en el organismo gubernamental o adquirido, debe tener un manual de uso, el cual sirva de soporte al usuario. Ver **sección 2.03 Documentación del software**.
  - (iv) El departamento de TIC debe dar soporte al usuario por medio de la mesa de servicio, correo electrónico o cualquier otro medio para los fines, cuando se experimente inconvenientes con el uso del software.
  - (v) Todo software licenciado, adquirido o desarrollado por el organismo, debe cumplir con las directrices establecidas en la **sección 1.04 Licenciamiento**.
  - (vi) El departamento de TIC debe elaborar políticas para el uso del software utilizado, en donde se establezcan los derechos y restricciones que tienen los usuarios.
    - a) Las políticas de uso del software deben ser enviadas a los usuarios mediante correo electrónico, Intranet o cualquier medio de comunicación que el organismo considere.
  - (vii) Los departamentos de TIC deben implementar controles que eviten la instalación y desinstalación de software sin autorización.
  - (viii) El software utilizado por el organismo debe estar provisto con el soporte necesario por parte del proveedor.



### Sub-sección 2.01.5.

### Políticas de administración del software

- (a) Para la correcta administración del software, el organismo debe seguir las siguientes directrices:
  - (i) El departamento de TIC debe realizar un inventario del software utilizado en el organismo, a fin de regularizar el uso de aplicaciones licenciadas y tener una mejor administración de los mismos. Ver **Sub-sección 3.04 sobre inventario general de TIC.**
    - a) En caso de que el departamento de TIC encuentre un software no licenciado en terminales o servidores de la infraestructura TIC, este debe comenzar un proceso de regularización del mismo; esto consiste en dos (2) procesos:
      - i) **Eliminación del software:** Consiste en la eliminación del software, en caso de que el mismo no sea un software de soporte, necesario para desempeño de los usuarios o servidores de la infraestructura TIC.
      - ii) **Regularización del software:** Consiste en la adquisición del software, en caso de que el mismo sea necesario para desempeño de los usuarios o a servidores de la infraestructura TIC. Ver **NORTIC A1:2014, sección 3.02 Compra y contratación de TIC.**
    - b) Como resultado del inventario, el departamento de TIC debe tener un listado del software instalado por terminal, con lo que debe determinar lo siguiente:
      - i) **Optimización de memoria:** Proceso en el cual el departamento de TIC determina si la terminal necesita un aumento de memoria para su correcto desempeño.
        - En caso de que la terminal sea transferida de un usuario a otro, la misma puede aplicar para una disminución de memoria, en caso de que el nuevo usuario no requiera de tanta capacidad.



- ii) **Optimización de espacio:** Proceso en el cual el departamento de TIC determina si la terminal necesita un aumento de espacio en almacenamiento.
  - Cuando la terminal sea transferida de un usuario a otro, la misma puede aplicar para una disminución de espacio de almacenamiento, en caso de que el nuevo usuario no requiera tanta capacidad.
- iii) **Eliminación del software no necesario para el desempeño del usuario:** Proceso en el cual el departamento de TIC determina qué software no es parte de las utilidades que el usuario necesita para el desempeño de sus funciones.

#### Sub-sección 2.01.6. Manejo de errores e incidentes del software

---

- (a) El departamento de TIC debe tomar en cuenta las siguientes directrices para el correcto manejo de errores e incidentes en el organismo.
  - (i) Todo error o incidente debe ser comunicado al departamento de TIC, por medio de la mesa de servicio o algún otro medio de comunicación usado por el organismo.
    - a) Luego del registro debe informársele al cliente como mínimo:
      - El número de caso.
      - El tiempo de respuesta para que sea resuelto su caso.
      - Soporte técnico asignado para dar resolución al incidente.
    - b) Debe existir un proceso de escalamiento para los incidentes que no puedan ser resueltos por el técnico asignado. En estos casos deberá seguirse el protocolo establecido por el organismo. Ver **anexo No. 1. Matriz de escalamiento para la gestión de incidentes.**



- c) Las solicitudes de servicio para la solución de un error o incidente deben asociarse a un estado, el cual puede ser:
- **Registrado:** Cuando el incidente se registra, sin embargo, por razones propias del organismo no se inicia su resolución hasta que el área o unidad responsable lo determine.
  - **Activo:** Cuando la resolución del incidente se inicia, inmediatamente se registra el mismo.
  - **Suspendido:** Cuando el estado del incidente estuvo activo y por razones propias del organismo, o el área responsable, la resolución se detuvo.
  - **Resuelto:** Cuando el incidente se ha solucionado y especificado el procedimiento llevado a cabo para su resolución, pero no se ha confirmado con el cliente.
  - **Cerrado:** Cuando se haya completado el proceso de resolución, incluyendo la confirmación con el cliente.
- d) Una vez resuelto el incidente, el departamento de TIC debe:
- Confirmar y notificar con el cliente la correcta resolución del incidente.
  - Registrar el procedimiento que se llevó a cabo para la resolución del incidente.
  - Cerrar la solicitud.
- (ii) Todo error o incidente debe seguir el siguiente nivel de escalamiento (ver anexo No. 2. Matriz de escalamiento y solución de errores e incidentes.) en caso de que el mismo no encuentre solución:
- a) **Nivel 1:** Línea de servicio de TIC.
  - a) **Nivel 2:** Gerencia del departamento TIC.
  - b) **Nivel 3:** Desarrolladores y especialistas del departamento TIC.





- c) **Nivel 4:** Proveedores, si aplica.
- (iii) El departamento de TIC debe buscar la mejora continua en su plan de gestión y manejo de errores e incidentes cumpliendo lo siguiente:
- Mejorar la productividad de los usuarios.
  - Cumplimiento de los Acuerdos de Niveles de Servicio (SLA, por sus siglas en inglés) acordados con el usuario.
  - Mayor control de los procesos y monitorización de los servicios.
  - Optimización de los recursos disponibles.
  - Una base de datos de conocimiento más precisa para el tratamiento de los incidentes.
  - Mejora de la satisfacción general de los usuarios y ciudadanos.

## SECCIÓN 2.02. Desarrollo del software gubernamental

---

En el Estado Dominicano son necesarias directrices y políticas para el correcto establecimiento de estándares sobre el software desarrollado en los organismos, para lo cual, en esta sección, se establecen las pautas necesarias para lograr un desarrollo óptimo bajo las mejores metodologías y prácticas. Ver **sección 2.05. Marco de desarrollo recomendado.**

### Sub-sección 2.02.1.

### Usabilidad

---

- (a) Para el desarrollo de un software que permita su fácil uso, debe tomarse en cuenta las siguientes directrices:
- (i) Debe evitarse el uso de ventanas emergentes sin solicitud del usuario.
  - (ii) Si la pantalla en la que se encuentra el usuario tiene un límite de tiempo, este debe informar al usuario antes de que expire la sesión.



- (iii) Debe tener una resolución de pantalla mínima de 1024 x 768 pixeles.
- (iv) Para la correcta implementación de un software usable, deben tomarse los siguientes puntos a consideración:
  - a) **Sencillez:** Que el software carezca de complejidades innecesarias.
  - b) **Indexabilidad:** Que el software indexe fácilmente los datos introducidos por el usuario, ya sean documentos o contenidos introducidos al sistema para ser fácilmente encontrados por una herramienta de búsqueda.
  - c) **Compatibilidad:** Que el software sea desarrollado para la arquitectura o plataforma requerida, y en caso de que la misma deba ser multiplataforma, esta sea desarrollada para la web o para las plataformas necesarias.
  - d) **Actualización:** Que el software sea actualizado al momento de detectar un fallo de seguridad y errores en el desempeño de la misma. De igual manera el software debe tener una herramienta de reportes de errores al desarrollador.
  - e) **Rapidez:** Una de las características más importantes, ya que esto puede definir la disminución de tiempo en los procesos y tareas que el usuario tiene asignado.
- (v) El software debe mantener informado al usuario en todo momento, cumpliendo con lo siguiente:
  - a) Debe destacarse el menú donde se encuentra el usuario.
  - b) El cursor debe cambiar de forma al pasar por encima de un botón o enlace en caso de una aplicación web; en caso de ser una aplicación de escritorio debe cambiar de apariencia el control.
  - c) El software debe brindar burbujas de ayuda para soporte al usuario.
  - d) En caso de que el software administre un proceso, debe retroalimentar al usuario.



- (vi) El software debe tener un lenguaje simple y llano, y utilizar términos técnicos cuando sea única y estrictamente necesario.
  - a) El software no debe tener un lenguaje informal.
  - b) El software no debe tener faltas ortográficas.
  - c) Cuando en el software haga uso de una sigla, esta debe estar desglosada en todas las palabras que la componen.
  - d) En caso que aplique, el software debe tener la posibilidad de ver diferentes traducciones del mismo.
  - e) Cuando el software muestre datos al usuario, los mismos deben estar presentados según el estándar del Sistema Internacional de Unidades.
- (vii) Cuando el software permita obtener reportes, estos deben cumplir con lo siguiente:
  - a) Permitir la posibilidad de imprimir el contenido.
  - b) Permitir la posibilidad de exportar el contenido a formatos abiertos.
- (viii) En caso del software brindar documentos descargables, debe proporcionar un medio para visualizarlos, o en su lugar un enlace para descargar un visor.
- (ix) El software debe tener un manual fácilmente visible que muestre toda la documentación de uso pertinente.
- (x) El software debe mantener una consistencia en el estilo gráfico, tomando en cuenta lo siguiente:
  - a) Encabezados principales.
  - b) Botones.
  - c) Listas.
  - d) Formularios y sus campos.
  - e) Alertas.
- (xi) Todas las alertas que el software presente, deben aparecer en el lugar del evento.
- (xii) El software debe proporcionar ayuda al usuario en todo momento:



- a) Cuando existan procesos o formularios, el software debe tener un documento guía, el cual debe presentarse al inicio del proceso.

### Sub-sección 2.02.2.

### Accesibilidad

- (a) Para el desarrollo de un software con niveles de accesibilidad óptimos, debe tomarse en cuenta las siguientes directrices:
  - (i) En caso del software ser web, y este sea de consumo al ciudadano, debe tener una versión móvil.
  - (ii) El software debe estar disponible en todo tiempo, 24 horas al día, 365 días del año.
  - (iii) El software no debe tener secciones sin contenido o funcionalidades.
  - (iv) El software no debe estar desarrollado en tecnologías discontinuadas.
  - (v) El software debe ejecutarse correctamente y de manera homogénea en todas las plataformas para la cual fue desarrollada.
  - (vi) El software debe permitir la navegación, a través del menú principal, y seleccionar todos los elementos del menú, incluyendo el menú del sistema por medio del teclado.
  - (vii) Cada función de la barra de herramientas debe ser seleccionable desde el teclado.
  - (viii) Todas las funciones básicas del teclado y atajos deben estar disponibles.
  - (ix) El usuario debe navegar por el área de texto o contenido por medio del teclado.
  - (x) El usuario debe acceder a cualquier interfaz de la aplicación utilizando el teclado.
  - (xi) Cuando se utilice elementos de multimedia en la aplicación, estos deben incorporar:
    - Una indicación de la señal del sonido y audio.
    - Una opción donde el usuario pueda habilitar o



- deshabilitar el sonido o ajustar el volumen.
- (xii) No debe utilizarse objetos que provoquen parpadeos o movimientos continuos en el contenido.
  - (xiii) La información o contenido no debe depender de los colores.
  - (xiv) No debe haber elementos escondidos u ocultos.
  - (xv) Todos los controles deben estar con una etiqueta que describa su utilidad.
  - (xvi) Debe evitarse que los controles no disponibles puedan ser marcados con el cursor.
  - (xvii) Todos los elementos de una ventana deben tener una tabulación y orden lógico.
  - (xviii) Todo tipo de fuente debe ser legible y entendible por el usuario.
  - (xix) Las imágenes e íconos deben ser descriptivos y objetivos a su función.

### Sub-sección 2.02.3.

### Interoperabilidad

- (a) Todo software desarrollado debe ser interoperable (ver **anexo No. 3. Modelo de implementación de interoperabilidad (MINT)**), para esto deben tomarse en cuenta las siguientes directrices:
  - (i) Todo software debe ser desarrollado en base a estándares abiertos, y los mismos deben cumplir las siguientes cualidades:
    - **Disponibilidad:** Deben estar disponibles para su lectura e implementación.
    - **Capacidad de elección:** Debe ser posible elegir la implementación a usar del mismo, sin restringir al cliente, un distribuidor o grupo concreto.
    - **Sin prebendas:** El software debe estar disponible sin coste alguno.
    - **Sin discriminación:** La elección de una implementación debe ser por motivos puramente técnicos.
    - **Extensión o reducción:** Las implementaciones pueden ser ampliadas o utilizar sólo un subconjunto del estándar.

- **Sin prácticas abusivas:** Su implementación debe evitar tácticas subversivas y cualquier acción que atente contra la privacidad de los usuarios.
- (ii) Las bases de datos con informaciones básicas de ciudadanos y los servicios que estos reciben, ya sean que estén desarrolladas por el organismo o por terceros, deben incluir una Interfaz de Programación de Aplicaciones (API<sup>[18]</sup>, por sus siglas en inglés) que permita a otros programas acceder a las informaciones contenidas en ellas.
- (iii) Los códigos fuente utilizados como base informativa a servicios electrónicos, deben incluir las API necesarias para lograr la interoperabilidad entre los sistemas institucionales del Estado Dominicano.
- (iv) Todo software debe cumplir con los protocolos mínimos para transferencia de datos, como se especifica a continuación:
  - a) Para la transferencia de archivos entre servidores o estaciones de trabajo, debe utilizarse los siguientes protocolos:
    - i) El Protocolo de Transferencia de Archivo Seguro (SFTP<sup>[19]</sup>, por sus siglas en inglés) utilizado con el Intérprete de Órdenes Seguras (SSH<sup>[20]</sup>, por sus siglas en inglés).
    - ii) El Protocolo Seguro de Transferencia de Datos (también referido como FTPS<sup>[21]</sup>, por sus siglas en inglés).
  - b) Para hipertexto y recursos de software debe utilizarse los siguientes protocolos:

[18] Son los procedimientos, funciones o servicios que se encuentran disponibles en determinadas librerías y/o bibliotecas para ser utilizados por otro software.

[19] Es un protocolo de red de utilizado para acceder y manejar archivos de manera remota utilizando métodos de encriptación.

[20] Es un protocolo y aplicación por el cual se accede remotamente a una computadora a través de una red de comunicación.

[21] Es un protocolo utilizado para la transferencia de archivos, el cual utiliza las propiedades de seguridad brindadas por el SSL para la comunicación.



- i) El Protocolo de Transferencia de Hipertexto (HTTP<sup>[22]</sup>, por sus siglas en inglés).
- ii) El Protocolo Seguro de Transferencia de Hipertexto (HTTPS<sup>[23]</sup>, por sus siglas en inglés).
- c) Para mensajería debe implementarse los siguientes protocolos, en su forma segura:
  - i) El Protocolo de Oficina de Correo (POP3<sup>[24]</sup>, por sus siglas en inglés), utilizado con uno de los siguientes protocolos de seguridad:
    - Protocolo de Seguridad de la Capa de Transporte (TLS<sup>[25]</sup>, por sus siglas en inglés) en el puerto 110.
    - Protocolo de Capa de Conexión Segura (SSL<sup>[26]</sup>, por sus siglas en inglés) en el puerto 995.
  - ii) El Protocolo de Acceso a Mensajes de Internet (IMAP, por sus siglas en inglés), utilizado con uno de los siguientes protocolos de seguridad:
    - Protocolo TLS en el puerto 143.
    - Protocolo SSL en el puerto 993.
  - iii) El Protocolo Simple de Transferencia de Correos (SMTP<sup>[27]</sup>, por sus siglas en inglés), utilizado con uno de los siguientes protocolos de seguridad:
    - Protocolo TLS en el puerto 587.
    - Protocolo SSL en el puerto 465.

[22] Es un protocolo utilizado para realizar la transferencia de texto o hipervínculos, a través de la web.

[23] Es un protocolo utilizado para realizar la transferencia de texto o hipervínculos de manera segura mediante el uso de cifrado, a través de la web.

[24] Es un protocolo utilizado de manera local para la obtención de correos electrónicos almacenados en un servidor remoto.

[25] Seguridad de la Capa de Transporte (TLS), Se encarga de proporcionar privacidad e integridad de datos entre dos aplicaciones que se comunican a través del internet.

[26] Es un protocolo que permite comunicación por medio de una red de forma segura.

[27] Es un protocolo simple de envío de correos electrónicos.



- (v) Todo software debe cumplir las siguientes directrices para la correcta exportación de datos:
- Notación de Objetos de JavaScript (JSON<sup>[28]</sup>, por sus siglas en inglés) y sus variantes.
  - Lenguaje de Marcas Extensible (XML<sup>[29]</sup>, por sus siglas en inglés) y sus variantes.
  - Valores Separados por Comas (CSV<sup>[30]</sup>, por sus siglas en inglés).
  - Valores Separados por Tabulaciones (TSV<sup>[31]</sup>, por sus siglas en inglés).
- (vi) Todo software debe cumplir los siguientes puntos para la presentación de archivos planos:
- JSON (y sus variantes).
  - XML (y sus variantes).
  - CSV.
  - TSV.
- (vii) Para el uso de extensiones y almacenamiento de datos debe seguirse las siguientes directrices:
- a) Puede utilizarse todas las extensiones existentes en el sector tecnológico (sin excepción) para los archivos generados, compilados o interpretados, siempre y cuando estos archivos no contengan códigos de software malicioso, programas espías o que comprometa la seguridad de los sistemas y/o servidores del organismo.
  - b) Debe documentarse el uso de cada extensión por cada sistema desarrollado o implementado y categorizado como sigue:

[28] Es un formato ligero usado como alternativa al XML para intercambio de datos.

[29] Es un lenguaje desarrollado por el Consorcio World Wide Web (W3C) para almacenar datos en forma legible. Este es utilizado para el intercambio de información entre diferentes plataformas.

[30] Es un formato de archivo de datos que su contenido está separado por comas.

[31] Es un formato de texto simple utilizado para el almacenamiento de información en forma de tablas. En este, cada registro de la tabla representa una línea del archivo de texto.





- Archivos de documentos.
  - Archivos de configuración.
  - Archivos de recursos.
  - Archivos temporales.
- c) Para el almacenamiento de manera local de los datos, los servidores del organismo deben cumplir con lo siguiente:
- i) La tecnología utilizada en los servidores para el almacenamiento de archivos por el organismo debe permitir la generación de accesos externos, mediante servicios web o vía HTTPS permitiendo la integración directa y segura con otros sistemas.
  - ii) Almacenar los archivos en servidores, donde el sistema operativo provea seguridad a la información.
  - iii) Los nombres de los archivos deben obedecer a un esquema o patrón definido por el organismo, el cual debe estar presente en la documentación del sistema.
  - iv) Los nombres de los archivos no deben ser mayor de 64 caracteres.
- d) Para el almacenamiento de los datos en la nube, el software debe cumplir con las directrices especificadas en la **NORTIC A1:2014, sección 4.03. Computación en la nube.**
- (viii) Todo software debe cumplir las siguientes directrices para la interoperabilidad web entre sistemas, los cuales deben estar basados en servicios web, utilizando al menos uno de los protocolos de transferencia estándares en la web, como lo son:
- JSON- Llamada a Procedimiento Remoto (JSON-RPC, por sus siglas en inglés).
  - JSON- Protocolo de Servicio Web (JSON-WSP<sup>[32]</sup>, por sus siglas en inglés).

[32] Es un protocolo de servicio Web usado por JSON para la descripción de servicios, respuestas y solicitudes.



- Transferencia de Estado Representacional (REST, por sus siglas en inglés).
  - Protocolo de Acceso de Objeto Simple (SOAP<sup>[33]</sup>, por sus siglas en inglés).
  - Servicio de Procesamiento Web (WPS, por sus siglas en inglés).
  - Lenguaje de Descripción de Servicios Web (WSDL, por sus siglas en inglés).
- (ix) Para lograr la interoperabilidad organizacional (ver NORTIC A4:2014, Capítulo III. Interoperabilidad organizacional) deben realizarse acuerdos intraorganizacionales o interorganizacionales, por lo que se han definido seis (6) tipos de acuerdos:
- **Acuerdos técnicos:** Se incluyen formatos, protocolos, sistema de seguridad, así como la descripción y definición de los estándares seleccionados.
  - **Acuerdos semánticos:** Establece los datos, metadatos<sup>[34]</sup> e informaciones para una interpretación correcta de lo que se está intercambiando.
  - **Acuerdos organizacionales:** En estos acuerdos se incluyen las bases de las reglas de acceso y caducidad de información, servicios sobre las conexiones, autenticación, privacidad, seguridad de la información, entre otros.
  - **Acuerdos legales:** Se requiere el uso de una política claramente definida, basada y sustentada en la creación, determinación de especificaciones y reglas dentro de un marco legalmente viable y flexible que permita agilizar la implementación fluida de la interoperabilidad gubernamental. Es importante que en este acuerdo se determinen las mejores prácticas con bases preferiblemente legales para mantener la privacidad de la información que se comparte, al igual de cual o cuales de los organismos participantes le compete dichas responsabilidades.

[33] Es un protocolo estándar de comunicación entre dos objetos por medio de XML.

[34] Son un conjunto de información que describe las características de otra información. Es "datos sobre datos".



- **Acuerdos económicos:** Consisten en la forma de hacer sustentable la implementación de un sistema interoperable, en el cual se incluyen acuerdos de cooperación, formas de pago, servicios con costos, entre otros.
  - **Acuerdos culturales:** Estos son acuerdos intraorganizacionales, los cuales refieren a los mecanismos que se utilizarán para generar una cultura organizacional basada en el modelo de sistema de información interoperable. Estos acuerdos son un factor crítico para el éxito.
- (x) Para la interoperabilidad semántica (ver **NORTIC A4:2014, Capítulo IV. Interoperabilidad semántica**), debe tomarse en cuenta las siguientes directrices:
- a) La información intercambiada debe estar descrita en función de metadatos.
  - b) La estructuración de los metadatos para los elementos y atributos debe cumplir con los siguientes lineamientos generales:
    - **Expresividad:** Poseer los conceptos necesarios para una explicación real de lo que se desea expresar.
    - **Sencillez:** Debe ser simple para su fácil comprensión.
    - **Singularidad:** Cada concepto debe tener un significado único.
    - **Precisión:** Los conceptos deben estar definidos de forma concisa y exacta.
  - c) Para la descripción de los elementos de datos, debe tomarse en cuenta el siguiente esquema de datos:
    - i) **Nombre:** Define el nombre asignado para identificar coherentemente al atributo.
    - ii) **Fecha:** Especifica la fecha de un evento del atributo. La fecha debe especificarse siguiendo el estándar ISO 8601, utilizando el sistema horario de 24 horas y organizando el metadato de más a menos significativos como sigue: YYYY-MM-DD hh:mm:ss.



- iii) **Descripción:** Donde se explica de forma breve de qué trata el contenido del atributo.
- iv) **Organismo:** Donde se especifica el organismo responsable de la creación del atributo.
- v) **Formato:** Donde se describe la forma o restricciones de representación y contenido.
  - Para la definición del formato deben especificarse los siguientes atributos:
    - Tipo de dato.
    - Longitud mínima.
    - Longitud máxima.
    - Número máximo de decimales.
    - Valor mínimo permitido.
    - Valor máximo permitido.
- vi) **Valores permitidos:** Donde se identifican los valores que puede asumir o tener un atributo, en un dominio y rango respectivo.
- vii) **Identificador:** Clave unívoca que permite al usuario una forma fácil para colocar un nombre técnico resumido que identifique la naturaleza del metadato y con posibilidad de ser común con otros. Como resultado de esta asignación se podrá buscar la información de una forma más rápida, precisa y menos densa dentro del universo de datos.
- viii) **Estado:** En donde se especifica la etapa actual del elemento de datos. El estado debe definirse en una de las siguientes etapas:
  - **En definición:** El elemento de datos está conceptualizada a partir de una solicitud.
  - **En desarrollo:** Se encuentra en proceso de creación y realización de pruebas funcionales necesarias.
  - **Disponible:** Se encuentra publicado y listo para su uso.



- **Obsoleto:** Se utilizan en algunas aplicaciones, pero no son recomendados para implementaciones tecnológicas.
- ix) **Tipo:** Indica la clase o naturaleza de datos que se van a procesar.
- x) **Versión:** En donde se presenta el número de versión actual. Este metadato debe definirse utilizando dos números separados por un punto (.), iniciando siempre este valor en 1.0.
  - Para aquellas versiones donde los cambios realizados sean menores, debe actualizarse el valor de la versión en 1.x, siendo x un valor numérico iniciado en 1.
  - Para aquellas versiones donde los cambios sean mayores, debe actualizarse el valor de la versión en x.0, siendo x un valor numérico iniciado en 2.
- xi) **Alias:** En donde se define nombres alternativos por los cuales se le puede conocer al atributo.
- xii) **Validación:** en donde se especifican las reglas que deben ser aplicadas en la construcción o definición del atributo.
  - Las validaciones especificadas en el metadato deben ser utilizadas por los servicios que utilicen el atributo. En caso que no existan validaciones el valor por defecto de este metadato debe ser “No Disponible” (N/D).
- d) Debe definirse la estructura de los metadatos mediante el Vocabulario para Catálogo de Datos (DCAT<sup>[35]</sup>, por sus siglas en inglés).
- e) Debe utilizarse el Formato de Transformación Unicode de 8 bit (UTF-8<sup>[36]</sup>, por sus siglas en inglés), para la codificación de caracteres a usar para cada atributo.

[35] Es un estándar definido por el Consorcio World Wide Web (W3C) y diseñado para facilitar la interoperabilidad entre catálogos de datos publicados en la web.

[36] Es un formato estándar de codificación de caracteres capaz de representar cualquier carácter. Por sus características es recomendado para ser utilizado en la codificación de correos electrónicos y páginas web.

- (xi) Para la implementación de estándares abiertos debe tomarse en cuenta lo siguiente:
- **Disponibilidad:** Deben estar disponibles para su lectura e implementación.
  - **Capacidad de elección:** Debe ser posible elegir la implementación a usar del mismo, sin restringir al cliente, un distribuidor o grupo concreto.
  - **Sin prebendas:** Su implementación debe estar disponible sin coste alguno.
  - **Sin discriminación:** La elección de una implementación debe ser por motivos puramente técnicos.
  - **Extensión o reducción:** Las implementaciones pueden ser ampliadas o utilizar sólo un subconjunto del estándar.
  - **Sin prácticas abusivas:** Su implementación debe evitar tácticas subversivas y cualquier acción que atente contra la privacidad de los usuarios.
- (xii) Para el desarrollo de un software debe tomarse en cuenta el **anexo No. 4. Catálogo de estándares interoperables.**

#### Sub-sección 2.02.4. Metodología para el desarrollo del software

- (a) Para una correcta metodología del desarrollo del software, debe tomarse en cuenta las siguientes directrices. Ver **anexo No. 5. Metodología de desarrollo del software gubernamental.**
- (i) Para el desarrollo de software, debe tomarse en cuenta el ciclo de vida como sigue a continuación:
- **Necesidades:** Establece la especificaciones y requerimientos del software.
  - **Diseño:** Especifica como el software debe cumplir los objetivos y requerimientos.
  - **Desarrollo:** Se programa la solución requerida.
  - **Prueba:** Se realizan pruebas y controles de calidad.
  - **Lanzamiento:** Se pone a disponibilidad una versión estable del software.



- **Mantenimiento:** Se definen revisiones y mejoras periódicas.
- (ii) Para el proceso de gestión de los requerimientos, deben seguirse las siguientes directrices:
  - a) Debe elaborarse una lista ordenada con todos los requerimientos del software gubernamental suministrados por los interesados, especificando como mínimo:
    - Prioridad de cada requerimiento.
    - Validación y aceptación de cada requerimiento por los interesados.
    - Características, funcionalidades, requisitos, mejoras y correcciones que se vayan realizando sobre cada entrega del prototipo del software gubernamental.
  - b) La lista de requerimientos del software gubernamental debe contar con los siguientes atributos:
    - Descripción del requerimiento.
    - Número de orden del requerimiento.
    - Tiempo estimado para desarrollo del requerimiento.
    - Valor del requerimiento dado por la parte interesada.
  - c) La prioridad y los detalles de cada requerimiento solo deben ser actualizados o modificados a solicitud del departamento de TIC con aprobación de las partes interesadas.
  - d) El software gubernamental debe dividirse en prototipos, los cuales serán mostrados y entregados a los interesados.
- (iii) Para el proceso de la planificación del desarrollo, deben seguirse las siguientes directrices:
  - a) Debe realizarse un plan de desarrollo en base a un periodo de tiempo no mayor de un mes.



- b) El plan de desarrollo debe contener como mínimo los siguientes elementos:
  - Lista de requerimientos seleccionados para la entrega del prototipo.
  - Objetivo que se alcanzará en la entrega de cada prototipo.
  - Detalles que contendrá la entrega de cada prototipo.
  - Capacidad de desarrollo para la ejecución del desarrollo.
- c) Debe decidirse y evaluarse cuáles son los requerimientos que se tomarán en cuenta de la lista de requerimientos para la entrega de cada prototipo.
- d) Una vez seleccionados los requerimientos para la entrega del prototipo y estos sean aprobados por las partes interesadas, debe elaborarse:
  - i) Un diseño general de la arquitectura del software, utilizando uno de los siguientes patrones arquitectónicos:
    - Arquitectura en base a modelos y vistas, y cualquier otro que se adapte a este patrón, con el objetivo de separar los datos, las funcionalidades del software y la interfaz del usuario.
    - Arquitectura por n-capas, donde el software sea segregado en el número de capas necesarias.
  - ii) Un diseño preliminar de las interfaces de usuario, pantallas y cualquier otro elemento o componente del software.
- e) Los diseños elaborados deben ser incluidos dentro del plan de desarrollo y actualizados cada vez que se genere algún cambio.
- f) Debe especificarse el tiempo que se tomará para entregar cada prototipo.





- e) Debe informarse a las partes interesadas mediante una documentación o comunicación, cómo será ejecutado el desarrollo para cumplir con la entrega del prototipo.
- (iv) Para el proceso de organización del desarrollo, deben seguirse las siguientes directrices:
  - a) Una vez seleccionados los requerimientos para la entrega del prototipo, debe listarse, dividirse y seleccionarse los requerimientos con los cuales se trabajará diariamente.
  - b) La lista de requerimientos diarios debe ser actualizada constantemente durante la realización del desarrollo, mostrando lo siguiente:
    - Requerimiento completado.
    - Requerimiento pendiente.
    - Requerimiento actualizado.
    - Requerimiento eliminado.
  - c) Cuando uno de los requerimientos de la lista de desarrollo diario pasa a ser innecesario este debe ser eliminado.
- (v) Para el proceso de desarrollo diario, deben seguirse las siguientes directrices:
  - a) Durante el desarrollo diario cada participante en el desarrollo debe informar:
    - Desarrollo realizado el día anterior.
    - Desarrollo que realizará.
    - Impedimentos que puedan atrasar la ejecución del desarrollo diario.
  - b) El prototipo debe desarrollarse cumpliendo con los requerimientos obtenidos en la lista de requerimientos del software gubernamental y con el diseño general de la arquitectura del software.
  - c) El código fuente debe estar comentado. Ver **sección 2.03 Documentación del software**.



- d) Las variables deben ser nombradas de acuerdo a su función en el código fuente.
  - e) Todo el código fuente debe estar claramente tabulado.
  - f) Debe implementarse herramientas para el control de versiones del código fuente.
  - g) Debe realizarse pruebas de unidad en cada uno de los módulos, verificando que estos cumplan con los requerimientos del software.
  - f) Debe realizarse pruebas de integración, verificando que los módulos sean interoperables entre sí.
- (vi) Para el proceso de revisión del desarrollo, deben seguirse las siguientes directrices:
- a) Para la revisión del desarrollo debe realizarse una prueba general del prototipo del software, donde se compruebe su funcionamiento y estabilidad.
  - b) Debe verificarse cuáles funcionalidades de la lista de requerimientos del software se han completados y cuáles están pendientes.
  - c) Si la parte interesada añade nuevos requerimientos después de presentar el prototipo, la lista de requerimientos del software debe ser actualizada.
  - d) Los resultados de la prueba general del prototipo del software deben incluirse dentro del plan de desarrollo.
- (vii) Para el proceso de recapitulación del desarrollo, deben seguirse las siguientes directrices:
- a) Debe identificarse y ordenarse las actividades más importantes que salieron de forma efectiva y las posibles mejoras a realizarse.
  - b) Estas informaciones deben estar contenidas dentro del plan de desarrollo.



## SECCIÓN 2.03.

## Documentación del software

---

Esta sección establece los lineamientos sobre la documentación y el control de calidad que debe cumplir el software en su vida útil dentro de los organismos.

### Sub-sección 2.03.1.

### Documentación general del software

---

- (a) El software desarrollado o contratado debe contar con un manual de uso, en donde se detallen como mínimo las siguientes informaciones:
- Descripción del software, donde se especifica de manera detallada todos los aspectos generales del software, abordando los requerimientos para el funcionamiento, las características, versión de la aplicación, entre otras.
  - Instalación y configuración del software, donde se establezcan las pautas y lineamientos que los organismos deben implementar al momento de instalar, reinstalar o actualizar un software.
  - Funcionamiento, donde se describe todas las funciones disponibles que tiene el usuario para la administración del software.
  - Manejo de errores, donde se establezcan los pasos que debe realizar el usuario para manejar cualquier error que afecte el funcionamiento del software.
  - Preguntas frecuentes.
- (b) El software debe contar con un documento que contenga las siguientes informaciones:
- Lenguaje en el que está desarrollado.
  - Versión de la aplicación.
  - Tipo de licencia.
  - Información de los desarrolladores de la aplicación.



### Sub-sección 2.03.2.

### Documentación del Código

- (a) Debe utilizarse nombres descriptivos para clases, métodos, objetos u otros elementos.
- (b) Los comentarios del código deben ser concisos.
  - (i) Deben comentarse los distintos bloques de código, aplicando un criterio uniforme para cada nivel y siguiendo un modelo basado en los siguientes aspectos:
    - Incluir en cada clase una breve descripción, autor y fecha de última modificación.
    - Incluir por cada método, una descripción de su objeto y funcionalidades, así como de los parámetros y resultados obtenidos.
  - (ii) Los comentarios deben explicar, de manera breve, la funcionalidad de un método antes de su declaración.
    - a) No debe incluirse en el comentario la forma de cómo el método realiza su funcionalidad.
  - (iii) No debe comentarse el código para manejo de cambios.
  - (iv) Los comentarios en el código fuente deben mantenerse actualizados.
    - a) Si en algún momento la funcionalidad del código cambia, deben actualizarse los comentarios.
    - b) De cambiar la naturaleza del algoritmo, debe actualizarse inmediatamente el comentario asociado.
  - (v) Debe mantenerse el mismo estilo de formato y comentarios en el código fuente para permitir una mejor comprensión del lector.
  - (vi) No debe utilizarse palabras o frases indebidas en los nombres de entidades<sup>[37]</sup>, métodos o comentarios dentro del código fuente.

[37] Es la representación de un objeto el cual posee características y atributos propios.



### Sub-sección 2.03.3.

### Mantenimiento del software

- (a) El organismo debe tener un plan de mantenimiento del software.
- (b) El plan de mantenimiento del software debe cumplir con las siguientes etapas:
  - (i) **Mantenimiento perspectivo:** Donde se aplican métodos para el perfeccionamiento de software en términos de desempeño, sostenibilidad, eficiencia del procesamiento, entre otros.
  - (ii) **Mantenimiento de manejo de errores:** Donde se establecen los métodos de corrección de errores del software.
  - (iii) **Mantenimiento adaptativo:** Donde se aplican las adaptaciones de los cambios producidos en el entorno, lo que puede llevar a nuevos requerimientos.
  - (iv) **Mantenimiento preventivo:** Donde se establecen lineamientos para prever averías o desperfectos futuros.
- (c) El organismo debe mantener actualizada su documentación.

### SECCIÓN 2.04.

### Refactorización del código

Esta sección establece las directrices que deben cumplirse de manera general para la refactorización<sup>[38]</sup> del código fuente, con el objetivo de optimizar el código.

- (a) Cuando el código fuente sea refactorizado, el software no debe perder su funcionalidad.
- (c) El código debe estar escrito y en funcionamiento antes de realizar una refactorización.
- (d) El organismo debe elaborar un documento con los motivos y justificación de la refactorización.
- (e) Debe realizarse pruebas de funcionamiento después de realizar la refactorización, verificando que el correcto funcionamiento del software.

[38] Técnica utilizada para la reestructuración del código fuente, tomando el código existente y mejorándolo, alterando su estructura interna sin modificar su comportamiento.



- (f) La refactorización debe estar orientada a los siguientes aspectos:
- **Comprensibilidad:** Quedando el código más fácil para su comprensión.
  - **Simplicidad:** Haciendo menos complejo el código.
  - **Código Legible:** Donde el código este autocommentado y bien tabulado presentando un código sencillo de leer por un humano.
  - **Manteamiento del código:** Realizando metodologías donde se facilite el mantenimiento del software.
- (g) Debe utilizarse una herramienta que facilite la refactorización del código fuente, según el tipo de lenguaje que se esté utilizando.

## SECCIÓN 2.05.

### Marco de desarrollo recomendado

- Para el desarrollo de software en el Estado Dominicano, se recomienda el uso de Java en conjunto con el uso de Spring como marco de trabajo
- Conjunto de herramientas para implementación de proyectos de desarrollo de software público.
  - **Java Editor:** En Eclipse contamos con un editor Java multi-perspectiva el cual se adapta a cada uno de los tipos de proyectos que se pueden realizar en JavaSE y JavaEE. Cada una de estas perspectivas constan con sus propios asistentes de configuración para agilizar el proceso de creación de proyectos.
  - **Mobile Editor:** Es un editor para desarrollo de aplicaciones móviles.
  - **UML Editor:** Esta herramienta de modelados de objetos, permite editar el diagrama de las clases, los casos de usos y los desarrolladores pueden código Java a partir del diagrama creado en este editor.



- **ERD Editor:** Permite desarrollar el modelo lógico de datos y el modelo físico de estos, también los desarrolladores pueden generar scripts de generación de esquemas conforme a soluciones de DBMS.
- **Batch Editor:** Esta es una perspectiva especializada para crear Batch Jobs.
- Conjunto de herramientas de pruebas de proyecto.
  - **Unit Test:** Herramienta que permite diseñar casos de prueba basados en JUnit. Utilizando TestCase Generator, una herramienta gráfica y también DB unit test cases los desarrolladores puede crear casos de prueba.
  - **Test Automation:** Esta herramienta permite automáticamente la ejecución de los casos de prueba creados.
  - **Test Coverage:** Utilizando el toolkit de código abierto EMMA podemos realizar análisis de prueba de cobertura para los casos de pruebas correspondientes.
- Conjunto de herramientas para build del proyecto.
  - **Build:** herramienta que permite correr el proceso de build en el proyecto utilizando Maven como herramienta estándar de build de proyectos. Con la definición de dependencias y las funciones de administración de librerías cuales son fuertes características de Maven, los desarrolladores no necesitarán de administrar directamente las librerías, no obstante, podrán administrarlas a través del repositorio de librerías del sistema.
  - **CI (Continuous Integration):** con Hudson para la build y la integración continua, el CI server, se integra con el repositorio de código aplica la carga del código dentro del repositorio y ejecuta el proceso de building, llevando un registro de los casos de éxitos y de fallas.
- Herramienta de configuración de proyecto.
  - Mediante un repositorio Git los desarrolladores podrán administrar sus proyectos, documentándolos, realizando versiones de estos, etc.



- Se recomienda la utilización de PostgreSQL para el almacenamiento de datos.
- Se recomienda el uso de Asterisk para soluciones VOIP.



## CAPÍTULO III

### MIGRACIÓN DEL SOFTWARE

Dado los constantes avances en las TIC, los organismos deben ir adaptando y mejorando cada día sus procesos y plataformas tecnológicas, permitiendo que estas cumplan con los estándares nacionales adaptados a los nuevos tiempos. Debido a esto, se ha incluido en la NORTIC A6, el capítulo para la migración del software, en el cual se presentan las pautas a seguir a la hora de planificar y llevar a cabo la migración de la plataforma tecnológica, tomando en cuenta todos los aspectos que se ven afectado por esta.

#### SECCIÓN 3.01.

#### Directrices generales

- (a) Cada organismo debe capacitar, tanto al personal técnico como al personal administrativo, en las nuevas plataformas a migrar, antes, durante y después del proceso.
  - (i) Primero debe capacitarse al personal técnico, el cual debe estar preparado para resolver cualquier problema que presenten los usuarios.

#### SECCIÓN 3.02.

#### Planificación de la migración

Una buena planificación es la base para la gestión eficiente de la migración. Por tal razón se han integrado una serie de directrices, las cuales tienen como objetivo permitir que el proceso de migración que realice un organismo, sea lo menos traumático posible.

- (a) Antes de iniciar un proceso de migración, cada organismo debe realizar un plan para la migración del software.



- (i) El plan para la migración debe contener las siguientes especificaciones:
- **Tiempo de duración:** Donde se especifica el tiempo que durará el proceso de migración.
  - **Fecha inicial:** Donde se especifica la fecha en que iniciará el proceso de migración.
  - **Fecha final:** Donde se especifica la fecha en que finalizará el proceso.
  - **Recursos materiales y humanos:** En donde se especifica los recursos involucrados en la migración y el personal que será utilizado o afectado por la migración. Los recursos humanos identificados deben clasificarse de acuerdo como se muestra en la directriz 3.02.c.i.b y asignar las responsabilidades correspondientes dentro del proyecto de migración.
  - **Terceros:** En donde se especifica el personal subcontratado que será necesario en el proceso de migración. (Cuando aplique).
  - **Costos:** En donde se especifica los costos envueltos en todo el proceso de migración.
- (b) El organismo debe realizar la recolección de información para determinar el estado del organismo antes de iniciar con la migración.
- (c) Para el levantamiento de información debe tomarse en cuenta los siguientes aspectos:
- (i) Debe realizarse el levantamiento del personal.
- a) En el levantamiento del personal, cada organismo debe registrar todo el personal que se verá afectado durante la migración.
  - b) El personal debe ser dividido en:
    - Personal técnico.
    - Personal administrativo.
  - c) Cada organismo debe asignar las siguientes responsabilidades dentro del personal técnico:



- **Soporte técnico:** Quienes se encargan de atender directamente al usuario final durante y después del periodo de migración.
  - **Administración y servicios:** Se encargará de la instalación, configuración y mantenimiento de la plataforma operativa de la organización.
  - **Desarrollo:** Se encargarán del desarrollo de aplicaciones y soluciones necesitadas, en caso de que el desarrollo se realice de manera interna.
- (ii) Debe realizarse el inventario del hardware que será afectado por la migración.
- a) Debe tomarse en cuenta los puntos establecidos en la **sección 3.04. Inventario general de TIC al momento de realizarse el inventario de hardware.**
- (iii) Debe realizarse el inventario del software utilizado y que afectará la migración.
- b) Debe tomarse en cuenta los puntos establecidos en la **sección 3.04. Inventario general de TIC al momento de realizarse el inventario de software.**
- (d) Debe informarse, por medio del correo electrónico institucional, a todo el personal afectado de la migración a realizar.

### SECCIÓN 3.03. Aspectos técnicos durante la migración

---

Para evitar al máximo los posibles errores que puedan presentarse durante y después de una migración, es necesario que los organismos sigan las directrices sobre los aspectos técnicos a tomar en cuenta.

- (a) Debe definirse la migración a realizar en uno de los dos (2) tipos mencionados a continuación:
- (i) Cuando se migre solo una parte de la plataforma tecnológica, debe definirse como tipo de **migración parcial**. Este tipo de migración puede dividirse en dos (2) tipos diferentes:
    - a) **Migración de los servicios**, debe clasificarse de esta forma, en los casos donde solo se migren las aplicaciones de los servidores.



- b) **Migración de los usuarios**, debe clasificarse de esta forma cuando solo se migren aplicaciones de los usuarios finales que sean compatibles con las aplicaciones instaladas en los servidores.
  - i) Debe capacitarse a todo el personal administrativo que se vea afectado por la migración.
  - ii) El organismo debe informar al personal administrativo sobre las razones de la migración para evitar rechazos a las nuevas aplicaciones.
- (ii) Cuando la migración englobe tanto los servicios como las aplicaciones de los usuarios, debe definirse como tipo de **migración completa**.
  - a) Debido a la complejidad de este tipo de migración, debe elaborarse un plan bien detallado donde se especifiquen los pasos a seguir en caso de presentarse errores o incompatibilidades inesperadas.
  - a) Este tipo de migración debe realizarse en el siguiente orden:
    - Primero, la migración de servicios.
    - Segundo, la migración de usuarios.
- (b) El tiempo seleccionado para realizar la migración debe ser en horario no laboral, de manera que este no afecte la productividad.
  - (i) En caso de que la migración sea extensa, o tenga un alto riesgo de fallo, el mismo debe ser realizado en los horarios que el organismo determine como de menor flujo de actividad.

### Sub-sección 3.03.1.

### Prueba de la migración

- (a) Antes de realizar la migración en el ambiente de producción, el organismo debe probar el funcionamiento correcto de las aplicaciones a migrar.
- (b) El organismo debe seleccionar una de las metodologías de prueba para la migración que mejor se adapten a su condición de las mencionadas a continuación:
  - Migración en ambiente de prueba.



- Migración piloto.
- (i) La metodología seleccionada debe permitir al organismo determinar:
  - Posibles riesgos al realizar la migración.
  - Una definición correcta de la ejecución de la migración.
  - El software no migrable y posibles incompatibilidades.
- (ii) Cuando la metodología de prueba seleccionada sea la migración en ambiente de prueba, el organismo debe cumplir con lo siguiente:
  - a) Contar con un ambiente de prueba en la organización o poseer los recursos tecnológicos para crear uno de manera temporal.
  - b) Las pruebas deben realizarse en un entorno que asemeje la realidad operativa de la organización.
- (ii) Cuando la metodología de prueba seleccionada sea la migración piloto, el organismo debe cumplir con lo siguiente:
  - a) La selección de equipos que formarán parte del piloto debe realizarse en base a la necesidad del organismo.
  - b) Siempre que sea posible, deben seleccionarse aquellos equipos que comprometan en menor grado la productividad del organismo.
  - c) En caso de que el organismo no disponga de equipos tecnológicos suficientes, tales como servidores, deberán utilizar máquinas virtuales para simular los equipos pilotos.

### SECCIÓN 3.04.

### Inventario general de TIC

- (a) Todo organismo debe realizar un inventario ordenado, completo y actualizado de todos los activos que estén bajo la responsabilidad del departamento de TIC.
  - (i) El inventario general de TIC debe estar organizado en dos categorías principales:



- Activos físicos: Donde se registrarán todos los equipos de la infraestructura TIC, estaciones de trabajo, portátiles y demás.
  - Activos de información: Donde se registrará todo el software utilizado.
- (ii) La unidad de operaciones de TIC debe tener un personal que asuma la función de llevar a cabo todo el proceso de inventario. Este tendrá la responsabilidad de coordinar las tareas que deben desarrollarse como se muestra a continuación:
- a) **Levantamiento de inventario:** Registrar todos los bienes que forman el equipamiento tecnológico bajo el control del departamento de TIC. Esta fase del inventario se realizará en caso de que el organismo no haya realizado un inventario anteriormente.
    - i) El levantamiento de inventario debe realizarse siguiendo las directrices especificadas en la **NORTIC A1:2014, sub-sección 2.04.1 Levantamiento de inventario.**
  - b) **Actualizaciones de inventario:** Agregar al inventario nuevos bienes adquiridos por el departamento de TIC, igualmente eliminar los bienes que han salido de la responsabilidad del organismo.
    - i) Las actualizaciones de inventario deben realizarse siguiendo las directrices especificadas en la **NORTIC A1:2014, sub-sección 2.04.2 Actualización de inventario.**
  - c) **Control de inventario:** Revisar físicamente los bienes que se encuentran en el inventario.
    - i) El control de inventario debe realizarse siguiendo las directrices especificadas en la **NORTIC A1:2014, sub-sección 2.04.3 Control de inventario.**

## CAPÍTULO IV

### GESTIÓN E IMPLEMENTACIÓN DE BASE DE DATOS

En el siguiente capítulo, se contemplan todos los lineamientos y directrices acerca de la implementación, gestión y uso de base de datos en el Estado Dominicano, permitiendo así, que los organismos puedan usar de forma clara, sencilla y ordenada todos sus conjuntos de datos, los cuales se convierten en información reutilizable para el mismo organismo y la ciudadanía.

#### SECCIÓN 4.01. Implementación de base de datos

El proceso de instalación, configuración e implementación de una base de datos, son actividades significativas para el personal encargado de gestionar una base de datos, debido a esto, ese personal necesita del conocimiento, la experiencia y el cuidado para llevar a cabo dichas tareas. Es por eso que en esta sección se plasman todas las medidas necesarias para una correcta gestión de la base de datos dentro del organismo.

##### Sub-sección 4.01.1. Implementación de un sistema de gestión de base de datos

- (a) Todo organismo, antes de realizar la adquisición de un Sistema de Gestión de Base de Datos (DBMS<sup>[39]</sup>, por sus siglas en inglés), debe definir su estrategia en base a las siguientes directrices:
  - (i) Debe verificar que el DBMS, sea compatible con el Sistema Operativo (OS, por sus siglas en inglés) implementado en la plataforma.

[39] Es un sistema utilizado para el control, mantenimiento y administración de las bases de datos.



- (ii) El organismo debe asegurar que el proveedor del DBMS, provee el mantenimiento y soporte para las versiones actuales y las futuras.
  - (iii) Debe analizarse la cultura de la organización, en términos de generación de información, para identificar las operaciones críticas, respecto a la cantidad de información que se va almacenar en la base de datos.
  - (iv) El proveedor del DBMS, debe realizar una prueba de rendimiento y estos resultados deben asemejarse a lo requerido o especificado por el organismo.
  - (v) La prueba de rendimiento debe ser documentada.
  - (vi) El DBMS, debe soportar el número de usuarios y el tamaño de la base de datos del organismo.
  - (vii) Todo DBMS adquirido por el organismo debe contener, como mínimo, las siguientes herramientas:
    - Consolas de administración.
    - Medidores de rendimiento.
    - Consola para consultas, mediante el Lenguaje de Consultas Estructurado (SQL, en caso de que sea necesario).
    - Respaldos.
    - Recuperación.
  - (iii) El personal a cargo de la gestión de la base de datos debe cumplir con los roles especificados en la resolución 51-2013 sobre la estructura organizacional de los departamentos de TIC.
- (b) La arquitectura del DBMS debe ser seleccionada en base a uno de los siguientes criterios:
- (i) **DBMS empresarial:** Para organizaciones que manejan bases de datos de gran tamaño, debido al número de usuarios concurrentes y múltiples tipos de aplicaciones ejecutándose simultáneamente, o que presenten un nivel de estructura tipo A, de acuerdo con la resolución 51-2013 sobre estructura organizacional de los departamentos de TIC.





- a) Cuando se seleccione la arquitectura por este criterio, el servidor de base datos<sup>[40]</sup> debe ejecutar multiprocesamiento<sup>[41]</sup>, paralelismo<sup>[42]</sup> y clustering<sup>[43]</sup>.
- (ii) **DBMS de trabajo en grupo:** Para organizaciones medianas o que presenten un nivel de estructura tipo B, de acuerdo con la resolución 51-2013 sobre estructura organizacional de los departamentos de TIC.
- (iii) **DBMS personal:** Cuando la base de datos sea utilizada por un usuario en particular.
- (iv) **DBMS móvil:** Para cuando los organismos necesiten consultar datos y los mismos no estén dentro de la red local<sup>[44]</sup>.
- (c) Los organismos deben asegurar alta disponibilidad en sus DBMS, y la mismas debe cumplir con una de las siguientes estructuras:
  - (i) Arquitectura distribuida, para cuando los datos necesiten ser repartidos en los diferentes nodos<sup>[45]</sup> o estos necesiten ser independientes y autosuficientes.
  - (ii) Arquitectura de disco compartido, para cuando la organización necesite procesar altos niveles de información.
- (d) Las bases de datos utilizadas por las aplicaciones en el organismo, deben ser compatibles con la Conectividad Abierta de Bases de Datos (ODBC, por sus siglas en inglés).

[40] Es un conjunto de datos almacenados de manera ordenada y que guardan relación entre ellos para su uso posterior.

[41] Es cuando se utilizan recursos de uno o más procesadores simultáneamente para la realización de un proceso o tarea.

[42] Es cuando dos o más bases de datos trabajan de forma simultánea y de forma paralela.

[43] Son un conjunto de dispositivos conectados entre sí que comparten información a alta velocidades y se comportan como una sola unidad.

[44] Es una red de datos con un alcance geográficamente limitado.

[45] Hace referencia a un computador u ordenador conectado a una red de datos.



## Sub-sección 4.01.2. Diseño y normalización de base de datos

- (a) Debe realizarse un diseño conceptual de la base de datos, tomando en cuenta los siguientes aspectos:
- (i) Una identificación de:
    - Las entidades de la base de datos.
    - Relaciones entre las entidades.
    - Asociación entre entidades y relaciones.
  - (ii) Determinación de:
    - a) Los dominios de los atributos.
    - a) Los identificadores.
    - b) Las jerarquías de generalización<sup>[46]</sup>.
  - (iii) Elaboración del diagrama entidad - relación.
- (b) Para evitar la redundancia de los datos, problemas de actualización en las tablas<sup>[47]</sup> y proteger la integridad de los mismos, deben cumplirse las siguientes formas de normalización:
- (i) Los valores para cada uno de los atributos deben ser atómicos<sup>[48]</sup>.
  - (ii) Todos los atributos que no son claves, deben depender de la clave completa y no solo de una parte esta.
- Esto solamente se aplica a entidades que tienen claves compuestas, es decir, que las claves están formadas por más de un atributo.*
- (iii) Todos los atributos deben depender solo de una clave, y no de ningún otro atributo.

[46] Es el proceso de separar una entidad en varias sub-entidades relacionándolas con la entidad a la que se refieren.

[47] Es el lugar en donde se guardan los datos generados por los sistemas de información dentro de una base de datos.

[48] En base datos, son aquellos valores que no pueden separarse o descomponerse en partes más pequeñas.



- (c) Las bases de datos relacionales deben soportar los siguientes formatos para el intercambio de información:
- Lenguaje de Marcas Extensible (XML, por sus siglas en inglés) y sus variantes.
  - Valores Separados por Coma (CSV, por sus siglas en inglés).
  - Valores Separados por Delimitadores (TVS, por sus siglas en inglés).
- (d) Las bases de datos NoSQL, deben soportar Notación de Objetos de JavaScript (JSON, por sus siglas en inglés) y sus variantes.

#### Sub-sección 4.01.3.

#### Instalación de un sistema de gestión de base de datos

- (a) Procurar el manual técnico previsto por el proveedor del DBMS, donde deben indicarse los requisitos funcionales para la instalación.
- (b) Debe asegurarse que la versión del sistema sea la apropiada.
- (c) Deben verificarse los recursos de hardware como:
- Modelo del servidor.
  - Capacidad de disco.
  - Memoria del servidor.
- (d) Deben cumplirse los requerimientos de almacenamiento, los cuales guardan la información de la base de datos e índices, tomando en cuenta las siguientes estructuras:
- Los archivos binarios y los ejecutables.
  - La base de datos y las tablas del sistema.
  - Los archivos de bitácoras.
  - Los scripts<sup>[49]</sup> de inicio y control.
  - Los archivos de procesamiento de errores.

[49] Es un lenguaje de programación ejecutado por un intérprete, en este caso, el navegador.



- (e) Todo DBMS, debe tener soporte para la siguiente estructura de memoria:
- (i) Para caché de datos, en la cual se almacenan los datos necesitados por las aplicaciones.
  - (ii) Para caché de programas, en la cual se guardan las configuraciones para los procesos almacenados<sup>[50]</sup>.
- (f) Para la instalación de un DBMS, deben configurarse los parámetros siguientes:
- Configuración de la memoria.
  - Número de procesadores a utilizar.
  - Bases de datos, tablas e índices.
  - Habilitación de las características del DBMS, necesitadas por el organismo.

#### Sub-sección 4.01.4.

#### Estándares y procedimientos para la implementación del DBMS

- (a) Deben implementarse convenciones de nombres<sup>[51]</sup> para nombrar los objetos de la base de datos, tomando en cuenta lo siguiente:
- (i) El nombre los objetos debe ser claro y descriptivo, tomando en cuenta las limitaciones de longitud del DBMS.
  - (ii) Debe hacerse uso de un diccionario de datos para disminuir los caracteres.
  - (iii) Los nombres de las entidades, tales como las tablas, deben definirse si se nombrarán en singular o plural.
  - (iv) Debe definirse la utilización de mayúsculas o minúsculas solamente.
  - (v) Las palabras utilizadas para definir o identificar entidades o elementos deben separarse mediante un guion bajo.

[50] Son un conjunto de instrucciones almacenadas en una base de datos, destinado a realizar una función en particular.

[51] Son las diferentes reglas que se utilizan para la correcta utilización de los caracteres en las variables, funciones o cualquier otra entidad en el código fuente.



Ejemplo: organismo\_nombre.

- (b) Para la gestión de base de datos debe documentarse todos los procedimientos, respecto a la gestión y estos deben incluir:
- Una declaración de las políticas de información de la organización.
  - Guía para establecer la propiedad de los datos.
  - Reglas de objetos de la base de datos.
  - Políticas de administración de los metadatos.
  - Políticas de permiso a la base datos.
  - Políticas de respaldo y recuperación.
  - Políticas de pase a producción.
  - Políticas de monitoreo y gestión.
- (c) Deben documentarse todas las consideraciones para el desarrollo de software (ver **sección 2.02. Desarrollo del software gubernamental**) que accedan a la base de datos, y las mismas deben contener:
- La descripción de los métodos de acceso a las aplicaciones de la base de datos.
  - Los estándares de codificación SQL.
  - La guía de interpretación de los códigos de errores del DBMS.
- (d) Debe realizarse una definición de las políticas de acceso a la base de datos y la misma debe contener:
- (i) La expiración de la clave de usuario.
  - (ii) Los privilegios de acceso del usuario sobre los objetos.
  - (iii) Privilegios del usuario para realizar tareas administrativas.



### Sub-sección 4.01.5.

### Uso de base de datos no relacional

- (a) Cuando se utilice el Lenguaje de Base de Datos No Relacionales (NoSQL<sup>[52]</sup>, por sus siglas en inglés) en aplicaciones transaccionales, deben contemplarse los siguientes requerimientos:
- (i) Requerimientos de estructura o esquema:
    - Definición de las propiedades, tipos y restricciones.
    - Definición de las relaciones.
    - No variabilidad en el esquema, a menos que sea necesario.
  - (ii) Requerimientos de accesos de datos:
    - Coherencia.
    - Recuperación de todos los registros.
    - Accesos a las referencias cruzadas o tablas.
- (b) Cuando se utilice NoSQL en aplicaciones computacionales, deben contemplarse los siguientes requerimientos:
- (i) Requerimientos de estructura o esquema:
    - Definiciones de las propiedades, tipos y restricciones.
    - No variabilidad en el esquema, a menos que sea necesario.
  - (ii) Requerimientos de accesos de datos:
    - Acceso parcial a los registros.
    - Procesamiento en sentido de columnas o procesamiento vertical.
    - Accesos a las referencias cruzadas o tablas.
    - Coherencia, cualquier lectura debe presentar los datos con la última actualización.
- (c) Cuando se utilice NoSQL en aplicaciones web, deben contemplarse los siguientes requerimientos:

[52] Es un lenguaje de programación utilizado para acceder a bases de datos no relacionales.



- (i) Requerimientos de estructura o esquema:
  - Definición de las propiedades, tipos y restricciones.
  - La capacidad de cambiar de esquema en el tiempo, sin afectar a los registros existentes.
- (ii) Requerimientos de accesos de datos:
  - Acceso parcial a los registros.
  - Rápida ejecución de la Creación, Lectura, Actualización y Eliminación (CRUD, por su sigla en inglés).

## SECCIÓN 4.02.

## Gestión de base datos

Toda organización que utilice base de datos para organizar su información, requiere de un conjunto de administraciones que aseguren su adecuado funcionamiento. En ese sentido, la administración de base de datos no siempre es practicada adecuadamente, para lograr resultados óptimos o esperados. Por tal razón, en esta sección se contemplan todas las directrices para la correcta gestión de la base de datos.

- (a) Deben definirse cuáles son los datos que se almacenan en la base de datos y establecer pautas para mantener y manejar los datos una vez estén almacenados.
  - (i) Deben identificarse y clasificarse los requerimientos de información, tanto funcionales como no funcionales proveniente de los usuarios.
  - (iii) Los modelos de datos deben elaborarse de acuerdo a los procesos de negocio de la organización y tomando en cuenta lo siguiente:
    - Realización de un modelo de datos conceptual.
    - Realización de un modelo de datos lógico.
  - (ii) Debe establecerse las políticas de acceso a la información.
  - (iii) Debe identificarse a los dueños de la información y a sus usuarios.
  - (iv) Debe establecerse los estándares de control de datos.



- (d) Deben asignarse responsabilidades para el control de la operación del equipo y el sistema operativo en donde está instalado el DBMS, tomando en cuenta los siguientes criterios:
- Soporte y mantenimiento de los servidores u otros sistemas relacionados al DBMS.
  - Planeación de respuesta a contingencias y otros problemas relacionados.
  - Programación de scripts.
  - Cualquier otra responsabilidad que se considere necesaria.
- (b) La persona responsable de la operación del equipo y el sistema operativo en donde se encuentra el DMBS, debe cumplir con los roles expuestos en la resolución 51-2013 sobre la estructura organizacional de los departamentos de TIC.

#### Sub-sección 4.02.1.

#### Respaldo, recuperación y seguridad de una base de datos

- (a) Deben realizarse respaldos y recuperación de la base de datos y estas deben cumplir con lo siguiente:
- (i) Los respaldos deben ser completos, cuando se refiera a la base de datos en su totalidad o por periodos de tiempo, cuando se le realice a partes o segmentos de la base datos en un tiempo determinado.
- (ii) Las pruebas de recuperación deben realizarse de manera periódica, y que ese tiempo determinado no debe ser muy extenso, con el objetivo de verificar si los respaldos que se obtienen son confiables e íntegros.
- (b) La administración de la seguridad de una base datos debe contemplar las medidas y los permisos necesarios para las siguientes ejecuciones:
- La creación de objetos como: las bases de datos, tablas, vistas y procedimientos almacenados.
  - La alteración de la estructura de los objetos de la base de datos.





- El acceso a las tablas de sistema.
  - La consulta y modificación de los datos en las tablas.
  - La creación de funciones y tipos de datos.
  - La ejecución de procedimientos almacenados.
  - El cambio de parámetros de configuración.
  - La ejecución de tareas de respaldo, recuperación y administrativas.
- (c) Todas las medidas, permisos y políticas implementadas para el respaldo, la recuperación y la seguridad de una base de datos, deben estar documentadas y disponibles para una fácil consulta.

#### Sub-sección 4.02.2. Monitoreo y afinamiento de una base datos

- (a) El organismo debe contar con herramientas que aseguren la operación y el funcionamiento de la base de datos y esta debe contar con las siguientes características como mínimo:
- Indicadores de rendimiento: La Unidad de Procesamiento Central (CPU, por sus siglas en inglés), memoria física<sup>[53]</sup> y disco duro<sup>[54]</sup>.
  - Capacidad de identificar segmentos de código con problemas.
  - Disponer de informes de gestión.
  - Almacenamiento de informes de rendimiento.
- (b) La herramienta de monitoreo usada o adquirida por el organismo debe informar acerca de los siguientes estados:
- La carga de trabajo: Para verificar la demanda a la que es sometido el DBMS.
  - Volumen de trabajo: Para definir la capacidad del servidor para procesar datos, en términos de recursos de hardware.

[53] Es un tipo de memoria utilizada por los computadores para el almacenamiento de información.

[54] Es un dispositivo para el almacenamiento de datos a través del magnetismo u óptica.



- Los recursos: Para administrar el hardware y las herramientas de software como:
    - El Kernel<sup>[55]</sup> de la base de datos.
    - Los controladores de caché<sup>[56]</sup>.
    - Los discos duros.
  - La contención: Para cuando la carga a la que es sometido el DBMS es muy alta y se encuentran comprometidos todos los recursos del servidor.
- (c) Para el afinamiento de la base de datos deben aplicarse las siguientes prácticas:
- (i) Identificar que las tablas tengan los índices adecuados para responder de la manera correcta a las consultas de los usuarios.
  - (ii) Configurar adecuadamente la memoria y los cachés de datos y procedimientos.
  - (iii) Alinear la implementación de las bases de datos con la infraestructura de TIC existente.
  - (iv) Monitorear constantemente las bases de datos y aplicaciones.
  - (v) Implementar procedimientos de reorganización de las bases de datos.
  - (vi) Implementar procedimientos de actualización de las estadísticas de las bases de datos.

[55] También conocido como núcleo, es la parte principal o central de un sistema operativo.

[56] Es utilizado para validar los bloques de información que se copian en una parte específica de la memoria de un computador o dispositivo.

## CAPÍTULO V

### REPOSITORIO DE SOFTWARE DEL ESTADO DOMINICANO

En el siguiente capítulo se establecen las directrices a tomar en cuenta para publicar el código fuente en el Repositorio de Software del Estado Dominicano y su almacenamiento en el mismo.

#### SECCIÓN 5.01.

#### Publicación en el repositorio

- (a) Los organismos gubernamentales deben publicar el código fuente que estos desarrollen en el portal [www.softwarepublico.gob.do](http://www.softwarepublico.gob.do).
  - (i) El CAMWEB debe ser el responsable de publicar el código fuente o las personas quienes los miembros de este comité designen. El mismo debe estar conformado como lo indica la **NORTIC A2:2016, sección 5.01 Comité administrador de los medios web.**
- (b) Los organismos gubernamentales deben cumplir con las siguientes directrices para publicar su código fuente en el repositorio de software:
  - (i) El organismo debe disponer de un correo electrónico personalizado con su nombre de dominio y con un usuario denominado “softwarepublico”. A modo de ejemplo, el resultado sería de la siguiente manera: [softwarepublico@nombredeldominio.gob.do](mailto:softwarepublico@nombredeldominio.gob.do).
  - (ii) La administración de este correo debe estar bajo la responsabilidad del CAMWEB.
  - (iii) El uso de este correo debe limitarse únicamente para realizar el registro en el repositorio.



- (iv) El organismo debe seleccionar en el repositorio, la categoría que más se relacione con la aplicación a publicar.
- (v) Especificar el tipo de licenciamiento, como se muestra en la **sección 1.04 Licenciamiento**.
- (c) El código fuente debe ser estable, garantizando un correcto funcionamiento del mismo.
- (d) La aplicación debe ser suministrada con su código fuente y con instrucciones sencillas para la ejecución de la misma.
- (e) El código fuente que se carguen al repositorio debe estar desarrollado en base a licenciamiento libre.
- (f) La aplicación debe disponer de información técnica sobre su desarrollo, así como también información sobre cómo instalar y utilizar la aplicación, como se muestra en la **Subsección 2.03.1 Documentación general del software**.
- (g) El organismo debe encargarse de publicar y administrar las versiones en el repositorio.
- (h) Cuando el organismo realice una mejora a su código fuente, debe actualizarla en el repositorio.
- (i) Todo código fuente cargado al repositorio debe cumplir con los estándares especificados en la Norma para la Interoperabilidad entre los Organismos del Gobierno Dominicano (NORTIC A4:2014).
- (j) Todo código fuente desarrollado o adquirido por el Estado debe utilizar como fuente de sus datos almacenados un sistema de base de datos que puede ser relacional o NoSQL.

## SECCIÓN 5.02.

## Políticas de uso del repositorio

- (a) El organismo debe cumplir con las siguientes políticas de uso del repositorio:
  - (i) Las políticas de uso del repositorio deben ser dadas a los usuarios, mediante correo electrónico institucional, Intranet o cualquier medio de comunicación que el organismo considere.
  - (ii) El organismo debe tener total administración de su cuenta y contraseña por motivos de seguridad.



- (iii) El organismo debe mantener la calidad de las aplicaciones cargadas en el repositorio.
- (iv) El organismo que realice un nuevo desarrollo tomando una o más aplicaciones del repositorio, debe colocarlo en el mismo.
- (v) Al subir una o más aplicaciones al portal, el organismo debe garantizar ser el titular de todos los derechos de propiedad respecto a las mismas.

### SECCIÓN 5.03.

### Software de control de versión

Como el repositorio de la OPTIC se encuentra basado en GIT, en esta sección se establecen las directrices para el manejo de código fuente en GIT.

- (a) Para establecer un patrón de trabajo en GIT el organismo debe regirse bajo los siguientes requisitos:
  - (i) La rama o **“Branch”** principal debe llamarse **“Master”**.
  - (ii) Debe crearse una rama de desarrollo o **“Development”**, extraída de la principal en donde se integren todas las funcionalidades para prueba y colaboración entre los desarrolladores.
  - (iii) Cada versión completamente funcional de la solución debe estar separado por ramas dentro del repositorio GIT.
  - (iv) El código fuente solo debe estar disponible localmente en el computador de trabajo de los desarrolladores y en el repositorio remoto de GIT.
  - (v) Debe colocarse un comentario detallado en cada confirmación de cambio o **“Commit”** que se realice en el sistema.
  - (vi) El código fuente de todas las aplicaciones desarrolladas para cualquier organismo gubernamental debe estar guardado en su última versión en el repositorio GIT.
- (b) Cuando se requiera copiar todo el historial de un repositorio, el organismo debe clonar el mismo.



- (i) Cuando el repositorio sea privado el organismo debe autenticarse para poder clonarlo.
- (c) Para trabajar en base a un repositorio ya existente y crear otro, el organismo debe hacer una derivación o **“Fork”**.

#### SECCIÓN 5.04. Recomendaciones sobre el uso repositorio de software del Estado Dominicano

---

- Se recomienda que el organismo publique la rama **“Beta”** seguido de un número de secuencia para que la comunidad de desarrolladores del Estado pueda hacer aportes.
- Se recomienda que el organismo publique librerías con herramientas para el consumo de la comunidad de desarrolladores del Estado.



## GLOSARIO DE TÉRMINOS

---

### **AMBIENTE DE PRODUCCIÓN**

Es donde el personal del organismo realiza sus operaciones diarias e interactúa con el ciudadano. En este ambiente se maneja la información sensible del organismo y cualquier incidente puede afectar la productividad del mismo.

### **AMBIENTE DE PRUEBAS**

Es el espacio que se utiliza en el área de TIC para comprobar el funcionamiento de una nueva implementación, con el objetivo de determinar posibles errores y mejoras antes de pasar a producción.

### **ATÓMICOS**

En base datos, son aquellos valores que no pueden separarse o descomponerse en partes más pequeñas.

### **BASE DATOS**

Es un conjunto de datos almacenados de manera ordenada y que guardan relación entre ellos para su uso posterior.

### **BSD**

Berkeley Software Distribution o BSD (en español, «distribución de software berkeley») fue un sistema operativo derivado del sistema Unix nacido a partir de los aportes realizados a ese sistema por la Universidad de California en Berkeley.

### **CAPA DE CONEXIÓN SEGURA (SSL)**

Es un protocolo que permite comunicación por medio de una red de forma segura.

### **CLUSTERING**

Son un conjunto de dispositivos conectados entre sí que comparten información a alta velocidades y se comportan como una sola unidad.

### **CÓDIGO FUENTE**

Es un conjunto de instrucciones redactas en base a las reglas sintácticas de un lenguaje de programación para desarrollar un software determinado.



## CONTROLADORES DE CACHÉ

Es utilizado para validar los bloques de información que se copian en una parte específica de la memoria de un computador o dispositivo.

## CONVENCIONES DE NOMBRES

Son las diferentes reglas que se utilizan para la correcta utilización de los caracteres en las variables, funciones o cualquier otra entidad en el código fuente.

## DISCO DURO

Es un dispositivo para el almacenamiento de datos a través del magnetismo u óptica.

## DOCUMENTO OFFICE OPEN XML (DOCX)

Es un formato de archivo libre especial para documentos de texto con formato.

## EL SISTEMA DE LICENCIAS CREATIVE COMMONS

Nos permite dos combinaciones que resultan en una licencia libre<sup>[57]</sup>, estas son: Atribución (BY) y Atribución-Compartir Derivadas Igual (BY-SA). La primera de ellas no es Copyleft, la segunda sí. Podemos usar este tipo de licencias en nuestros manuales y artículos técnicos, no se suelen utilizar en la documentación adjunta al software sino para material complementario.

## ENTIDADES

Es la representación de un objeto el cual posee características y atributos propios.

## FORMATO DE DOCUMENTO PORTÁTIL (PDF)

Es un formato de almacenamiento de datos que funciona y puede ser visualizado independientemente de la plataforma, siendo así portátil y multiplataforma para su visualización.

## FORMATO DE TRANSFORMACIÓN UNICODE DE 8BIT (UTF-8)

Es un formato estándar de codificación de caracteres capaz de representar cualquier carácter. Por sus características es recomendado para ser utilizado en la codificación de correos electrónicos y páginas web.

[57] Basándonos en el Copyright podemos licenciar nuestro programa con condiciones que lo hagan libre. Los desarrolladores de software privativo emplean el copyright para quitar la libertad a los usuarios; nosotros empleamos ese copyright para garantizar su libertad.





## **GIT**

Es un sistema de control de versiones de código abierto que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, y estas versiones específicas pueden ser utilizadas más adelante.

## **HARDWARE**

Se refiere a todas las partes físicas o tangibles de un sistema de información.

## **HOJA DE CÁLCULO DE DOCUMENTO ABIERTO (ODS)**

Es un formato de archivo de estándar abierto especial para hojas de cálculo.

## **HOJA DE CÁLCULO OFFICE OPEN XML (XLSX)**

Es un formato de archivo libre especial para hojas de cálculo.

## **INTERFAZ DE PROGRAMACIÓN DE APLICACIONES (API)**

Son los procedimientos, funciones o servicios que se encuentran disponibles en determinadas librerías y/o bibliotecas para ser utilizados por otro software.

## **INTEROPERABILIDAD**

Es la capacidad que tiene un sistema de información para intercambiar datos con otros sistemas con la capacidad de procesarlos.

## **INTÉRPRETE DE ÓRDENES SEGURA (SSH)**

Es un protocolo y aplicación por el cual se accede remotamente a una computadora a través de una red de comunicación.

## **JERARQUÍAS DE GENERALIZACIÓN**

Es el proceso de separar una entidad en varias sub-entidades relacionándolas con la entidad a la que se refieren.

## **KERNEL**

También conocido como núcleo, es la parte principal o central de un sistema operativo.

## **LA LICENCIA DE LA FUNDACIÓN MOZILLA**

Cumple completamente con la definición de software de código abierto de la Open Source Initiative (OSI) y con las cuatro libertades del software libre enunciadas por la Free Software Foundation (FSF).



## **LA LICENCIA PÚBLICA ECLIPSE (EPL)**

Es una licencia de software de código abierto utilizada por la Fundación Eclipse para su software. Sustituye a la Licencia Pública Común (CPL) y elimina ciertas condiciones relativas a los litigios sobre patentes.

## **LA LICENCIA PÚBLICA GENERAL DE GNU**

Es una licencia libre y gratuita con derecho de copia para software y otros tipos de obras.

## **LENGUAJE DE CONSULTA ESTRUCTURADO (SQL)**

El SQL es un lenguaje de programación utilizado para acceder a bases de datos relacionales.

## **LENGUAJE DE CONSULTA PARA RDF (SPARQL)**

Es un lenguaje de consulta para grafos estandarizados en RDF.

## **LENGUAJE DE MARCAS EXTENSIBLE (XML)**

Es un lenguaje desarrollado por el Consorcio World Wide Web (W3C) para almacenar datos en forma legible. Este es utilizado para el intercambio de información entre diferentes plataformas.

## **LICENCIA LIBRE**

Basándonos en el Copyright podemos licenciar nuestro programa con condiciones que lo hagan libre. Los desarrolladores de software privativo emplean el copyright para quitar la libertad a los usuarios; nosotros empleamos ese copyright para garantizar su libertad.

## **MEMORIA FÍSICA**

Es un tipo de memoria utilizada por los computadores para el almacenamiento de información.

## **METADATOS**

Son un conjunto de información que describe las características de otra información. Es “datos sobre datos”.

## **MULTIPROCESAMIENTO**

Es cuando se utilizan recursos de uno o más procesadores simultáneamente para la realización de un proceso o tarea.



## **NODO**

Hace referencia a un computador u ordenador conectado a una red de datos.

## **NOSQL**

Es un lenguaje de programación utilizado para acceder a bases de datos no relacionales.

## **NOTACIÓN DE OBJETOS DE JAVASCRIPT (JSON)**

Es un formato ligero usado como alternativa al XML para intercambio de datos.

## **NOTACIÓN TURTLE**

Conocido también como Lenguaje de Notación 3 o “N3”, es un lenguaje utilizado para sintaxis XML de RDF.

## **ORGANIZACIÓN INTERNACIONAL DE NORMALIZACIÓN (ISO)**

Es una organización encargada de la creación de normas y estándares internacionales en diferentes áreas como tecnologías, seguridad, servicios, entre otros.

## **PARALELISMO**

Es cuando dos o más bases de datos trabajan de forma simultánea y de forma paralela.

## **PATRÓN DE DISEÑO**

Utilizado para la búsqueda de soluciones de problemas en el desarrollo del software, así como también para el diseño de interacción o interfaces.

## **PIXEL**

Es la unidad más pequeña que compone una imagen.

## **PORTAL WEB**

Es un conjunto de páginas electrónicas que presentan información y recursos de interés al usuario.

## **PROCESOS ALMACENADOS**

Son un conjunto de instrucciones almacenadas en una base de datos, destinado a realizar una función en particular.



### **PROTOCOLO DE ACCESO A OBJETOS SIMPLE (SOAP)**

Es un protocolo estándar de comunicación entre dos objetos por medio de XML.

### **PROTOCOLO DE OFICINA DE CORREO 3 (POP3)**

Es un protocolo utilizado de manera local para la obtención de correos electrónicos almacenados en un servidor remoto.

### **PROTOCOLO DE SERVICIO WEB JSON (JSON-WSP)**

Es un protocolo de servicio Web usado por JSON para la descripción de servicios, respuestas y solicitudes.

### **PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS SEGURO (SFTP)**

Es un protocolo de red de utilizado para acceder y manejar archivos de manera remota utilizando métodos de encriptación.

### **PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS/ CAPA DE CONEXIÓN SEGURA (FTP/SSL)**

Es un protocolo utilizado para la transferencia de archivos, el cual utiliza las propiedades de seguridad brindadas por el SSL para la comunicación.

### **PROTOCOLO DE TRANSFERENCIA DE HIPERTEXTO (HTTP)**

Es un protocolo utilizado para realizar la transferencia de texto o hipervínculos, a través de la web.

### **PROTOCOLO PARA LA TRANSFERENCIA SIMPLE (SMTP)**

Es un protocolo simple de envío de correos electrónicos.

### **PROTOCOLO SEGURO DE TRANSFERENCIA DE HIPERTEXTO (HTTPS)**

Es un protocolo utilizado para realizar la transferencia de texto o hipervínculos de manera segura mediante el uso de cifrado, a través de la web.

### **RED LOCAL**

Es una red de datos con un alcance geográficamente limitado.



## **REFACTORIZACIÓN**

Técnica utilizada para la reestructuración del código fuente, tomando el código existente y mejorándolo, alterando su estructura interna sin modificar su comportamiento.

## **SCRIPTS**

Es un lenguaje de programación ejecutado por un intérprete, en este caso, el navegador.

## **SEGURIDAD DE LA CAPA DE TRANSPORTE (TLS)**

Se encarga de proporcionar privacidad e integridad de datos entre dos aplicaciones que se comunican a través del internet.

## **SERVIDORES**

Son equipos informáticos que forman parte de una red de datos y que proveen servicio a otros equipos en dicha red, llamados clientes.

## **SISTEMA DE GESTIÓN DE BASE DE DATOS (DBMS)**

Es un sistema utilizado para el control, mantenimiento y administración de las bases de datos.

## **SISTEMA EMBEBIDO**

Son sistemas utilizados para controlar una o varias funciones, con recursos limitados y en condiciones no favorables.

## **SISTEMA OPERATIVO (OS)**

Es un software utilizado en los sistemas de información para gestionar y administrar los recursos de los dispositivos o computadores.

## **SOFTWARE**

Se refiere a todos los componentes lógicos o intangibles de un sistema de información, tales como programas, aplicaciones, sistemas operativos, entre otros.

## **SOFTWARE LIBRE**

El software libre es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas.



## **SOFTWARE PRIVATIVO**

El término ha sido creado para designar al antónimo del concepto de software libre, por lo cual en diversos sectores se le han asignado implicaciones políticas relativas al mismo.

## **TABLA**

Es el lugar en donde se guardan los datos generados por los sistemas de información dentro de una base de datos.

## **TEXTO DE DOCUMENTO ABIERTO (ODT)**

Es un formato de archivo libre especial para documentos de texto con formato.

## **USUARIO**

Hace referencia a la persona que consume o manipula un producto, servicio o información.

## **VALORES SEPARADOS POR COMAS (CSV)**

Es un formato de archivo de datos que su contenido está separado por comas.

## **VALORES SEPARADOS POR TABULACIONES (TSV)**

Es un formato de texto simple utilizado para el almacenamiento de información en forma de tablas. En este, cada registro de la tabla representa una línea del archivo de texto.

## **VOCABULARIO PARA CATÁLOGO DE DATOS (DCAT)**

Es un estándar definido por el Consorcio World Wide Web (W3C) y diseñado para facilitar la interoperabilidad entre catálogos de datos publicados en la web.



## ABREVIATURAS Y ACRÓNIMOS

No.	Abreviaturas y Acrónimos	Inglés	Español
1	API	Application Programming Interface	Interfaz de programación de aplicaciones
2	BSD	Berkeley Software Distribution	Distribución de Software Berkeley
3	CAMWEB	N/A	Comité Administrador de los Medios Web
4	CRUD	Create, Read, Update and Delete	Creación, Lectura, Actualización y Eliminación
5	CSV	Comma-Separated Values	Valores Separados por Coma
6	CSV	Comma-Separated Values	Valores Separados por Coma
7	DCAT	Data Catalog Vocabulary	Vocabulario para Catálogo de Datos
8	EPL	The Eclipse Public License	Licencia Pública Eclipse
9	FTPS	File Transfer Protocol / Secure Sockets Layer	Protocolo de Transferencia de Archivos / Protocolo de Capa de Conexión Segura
12	GLGPL	Library General Public License	Biblioteca General Pública
13	GNU GPL	General Public License	licencia General Pública



14	HTTP	Hypertext Transfer Protocol	Protocolo de Transferencia de Hipertexto
15	HTTPS	Hypertext Transfer Protocol Secure	Protocolo Seguro de Transferencia de Hipertexto
16	ISO	International Organization for Standardization	Organización Internacional de Normalización
17	JSON	JavaScript Object Notation	Notación de objetos de JavaScript
18	JSON-RPC	JSON - Remote Procedure Call	JSON- Llamada a Procedimiento Remoto
19	JSON-WSP	JSON - Web Service Protocol	JSON - Protocolo de Servicio Web
20	MIT	Massachusetts Institute of Technology	Massachusetts Instituto de Tecnología
21	MPL	Mozilla Public License	licencia Pública Mozilla
22	NoSQL	Nom Structured Query Language	Lenguaje de Base de Datos No Relacionables
23	ODBC	Open DataBase Connectivity	Conectividad Abierta de Bases de Datos
24	PDF	Portable Document Format	Formato de Documento Portátil
25	POP3	Post Office Protocol version 3	Protocolo de Oficina de Correo, versión 3





26	REST	Representational State Transfer	Transferencia de Estado Representacional
27	SFTP	Secure File Transfer Protocol	Protocolo Seguro de Transferencia de Archivos
28	SLA	Service Level Agreement	Acuerdos de Nivel de Servicio
29	SMTP	Simple Mail Transfer Protocol	Protocolo para la Transferencia Simple
30	SOAP	Simple Object Access Protocol	Protocolo de Acceso a Objetos Simple
31	SQL	Structured Query Language	Lenguaje de consulta estructurado
32	SSH	Secure SHell	Intérprete de Órdenes Seguras
33	SSL	Secure Sockets Layer	Capa de Conexión Segura
34	TIC	N/A	Tecnología de la Información y Comunicación
35	TLS	Transport Layer Security	Seguridad de la Capa de Transporte
36	TSV	Tab-Separated Values	Valores Separados por Delimitadores
37	TVS	Tab-Separated Values	Valores Separados por Delimitadores
38	UTF-8	8-bit Unicode Transformation Format	Formato de Transformación Unicode de 8-bit



39	WPS	Web Processing Service	Servicio de Procesamiento Web
40	WSDL	Web Services Description Language	Lenguaje de Descripción de Servicios Web
41	XML	Extensible Markup Language	Lenguaje de Marcas Extensible



## BIBLIOGRAFÍA

---

- Taller de Migración al Software Libre. España.
- Software libre en la administración pública: Desafíos y oportunidades.
- (2012, Noviembre ). Licencia de Software Público Gobierno de Chile.
- BSA, Business Software Alliance. Guía para la administración.
- Centro Nacional de Tecnologías de Información. Guía para el plan de migración a software libre en la administración pública nacional (APN) de la República Bolivariana de Venezuela. Caracas, Venezuela.
- Dirección de Estudios, Investigación y Estrategia de Gobierno Electrónico (DiGOB) de la Oficina Presidencial de Tecnologías de la Información y Comunicación (OPTIC). Software Público, Libre y de Código Abierto.
- Dirección nacional de tecnología informática. Normas de buen uso de hardware y software.
- Gobierno de Ecuador. (2007). Software Libre Ecuador. Quito, Ecuador.
- Gómez Labrador , R. (2005). Tipos de licencias de software. Estados Unidos.
- ISO 8601.
- ISO/IEC 27001. Implementación de Base de Datos.
- ITIL Foundation v3.
- MENDEZ, L., TINETTI, D., & GARRIDO, D. Refactoring de código estructurado.
- Ministerio de ciencia y tecnología-Oficina de tecnología de información. (2004). Libro amarillo del software libre. Caracas, Venezuela.



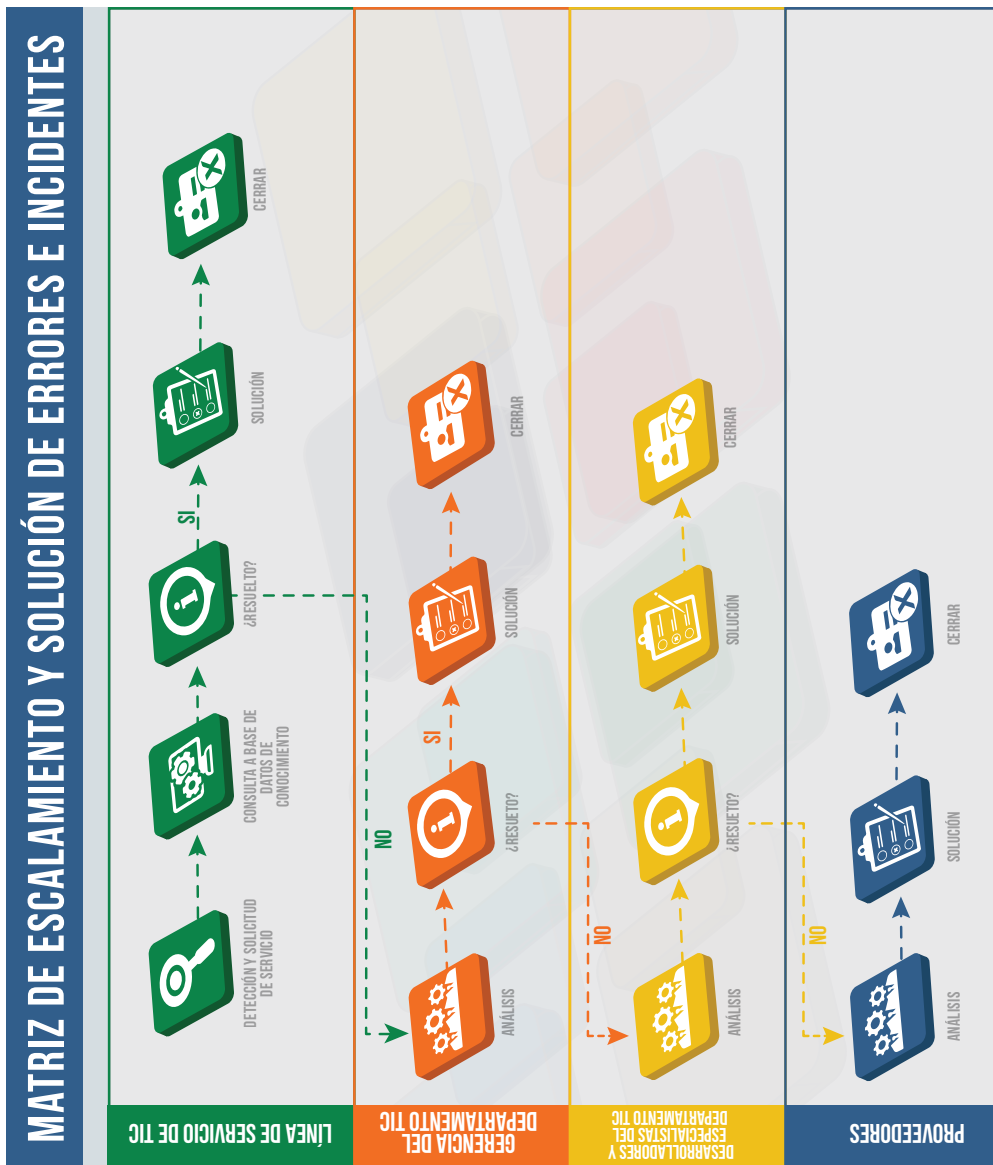
- Oficina Presidencial de Tecnologías de la Información y Comunicación (OPTIC). (2013). Norma General sobre Uso e Implementación de las Tecnología de la Información y Comunicación en el Estado Dominicano (NORTIC A1). Santo Domingo, República Dominicana.
- Oficina Presidencial de Tecnologías de la Información y Comunicación(OPTIC).(2014).NormasobreInteroperabilidad entre los Organismos del Gobierno Dominicano (NORTIC A4). Santo Domingo, República dominicana.
- OPTIC y Dominicana.gob.do. Política de software Público . República Dominicana.
- Presidencia de la República. (2004). Software Libre Venezuela. Caracas, Venezuela.
- Presidente Constitucional de la República de Ecuador. (2008). Software Libre Ecuador. Quito, Ecuador.
- UCLM Universidad de Castilla la Mancha. Servicio de Apoyo a la Investigación. Licencias para software libre. España.

## ANEXOS

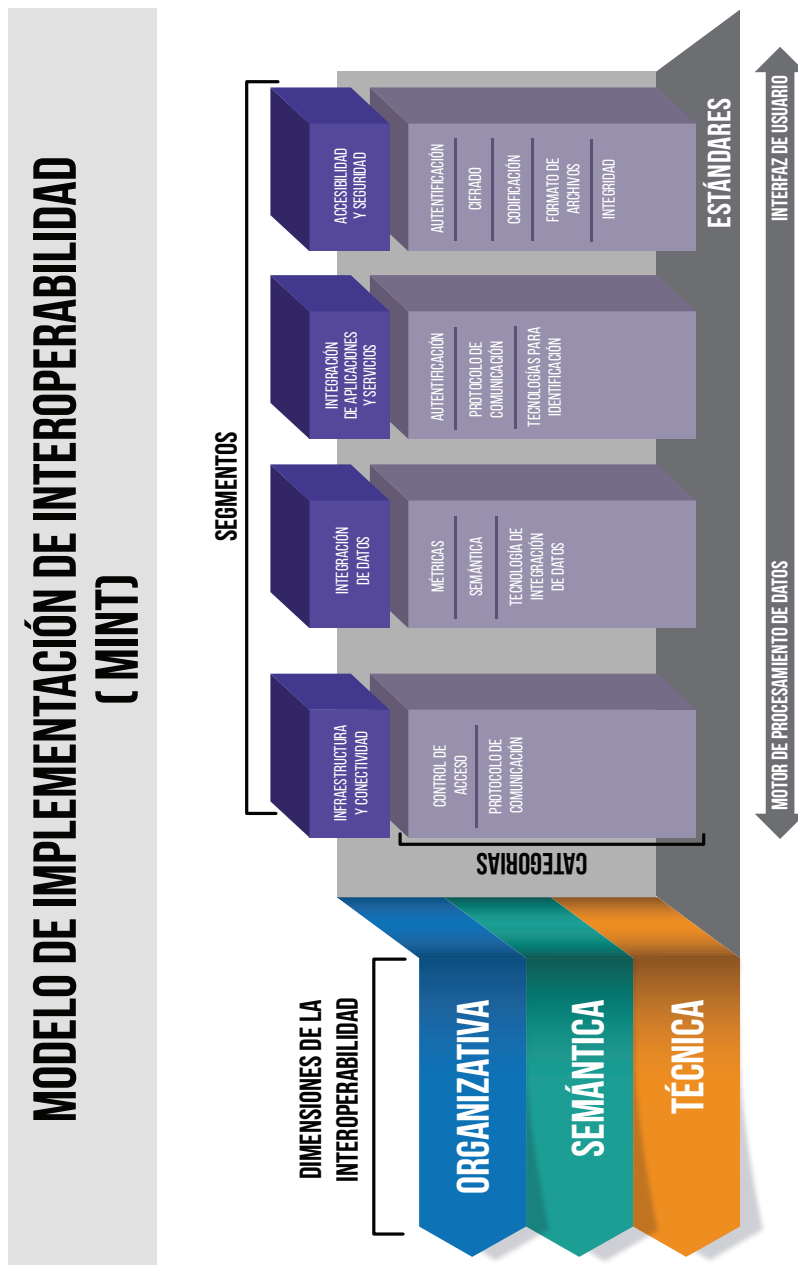
Anexo No. 1. Matriz de escalamiento para la gestión de incidentes.



Anexo No. 2. Matriz de escalamiento y solución de errores e incidentes.



Anexo No. 3. Modelo de implementación de interoperabilidad (MINT).





Anexo No. 4. Catálogo de estándares interoperables.

Segmentos de interoperabilidad	Categoría	Nombre		Tipo	Versión mínima aceptada	Extensión	Estado
		Común	Formal				
Accesibilidad y seguridad	Autenticación - Firma electrónica	XAdES	XML Advanced Electronic Signatures	Abierto	1.2.2	.xml .dsig .xsig	Estable
		XML-DSig	XML Signature Syntax and Processing	Abierto	Segunda edición 2008	.xml .dsig .xsig .sig	Estable
	Autenticación - Política Firma electrónica	ETSI TR 102 272	ETSI TR 102 272 Electronic Signatures and Infrastructures (ESI); ASN.1 format for signature policies	Abierto	RFC 3125 1.1.1	N/A	Estable
	Cifrado	SSH	Secure Shell	Abierto	1.99 (SSH 2)	N/A	Estable
		TLS	Transport Layer Security (TLS)	Abierto	RFC 5878 RFC 5746 RFC 5705 RFC 5489 RFC 5487 RFC 5469 RFC 5289 RFC 5288	N/A	Estable
Codificación - Codificación de caracteres	Base16, Base32 y Base64	The Base16, Base32 and Base64 Data Encodings	Abierto	RFC 4648	N/A	Estable	





	UCS UTF-8	ISO/IEC 10646:2003 Information technology -Universal multiple - Octet Coded Character Set	Abierto	2003		Estable
Codificación - Idioma	RFC 4646 ISO 639	RFC 4646 -Tags for Identifying Languages. ISO 639 - Codes for the Representation of Names of Languages	Abierto	2002 - 2008 RFC 4646	N/A	Estable
Formatos de archivos - Cartografía vectorial y SIG	GML	Geographic Markup Languages	Abierto	2007	.gml	Estable
	WFS	Web Feature Service	Abierto	2010	.gml	Estable
Formatos archivos- Compresión de archivos	GZIP	GNU ZIP	Abierto	RFC 1952	.gz	Estable
	RAR	Roshal Archive	Propietario	1.5	.rar .rev .r0x	Estable
	ZIP	ZIP RFC 1952	Abierto	N/A	.zip	Estable
	7ZIP	7ZIP	Abierto	9.0	.7zip	Estable
Formatos archivos- Imagen y/o texto	CSV	Comma Separated Values	Abierto	RFC 4180	.csv .txt	Estable
	HTML	HyperText Markup Language	Abierto	4.0.1	.html .htm	Estable
	CSS	Cascading Style Sheets	Abierto	2.1	.css	Estable
	JPEG / JPG	Join Photographic Experts Group	Abierto	ISO/IEC 10918- 4:1999, T.86 (06/98)	.jpg .jpeg	Estable



		ODF	Open Document Format	Abierto	1.0	.odt .ods .odp .odg	Estable
		Strict Open XML	Strict Open eXtensible Markup Lenguaje	Abierto	2012	.docx .xlsx .pptx	Estable
		PDF	Portable Document Format	Abierto	1.4	.pdf	Estable
		PNG	Portable Network Graphics	Abierto	ISO/IEC 15948,[1] IETF RFC 2083	.png	Estable
		SVG	Scalable Vector Graphics	Abierto	1.1	.svg	Estable
		TIFF	Tag Image File Format	Abierto	2004	.tiff	Estable
		TSV	Tab-separated Values	Abierto	N/A	.tsv .tab	Estable
		TXT	Texto Plano	Abierto	N/A	.txt	Estable
		XHTML	eXtensible HyperText Markup Language	Abierto	1.0	.html .htm	Estable
	Integridad	SHA	Secure Hash Algorithms	Abierto	RFC 4634 RFC3874	N/A	Estable
Infraestructura y conectividad	Control de acceso	LDAP	Lightweigh Directory Access Protocol.	Abierto	RFC 4510	N/A	Estable
	Protocolos de comunicación e intercambio - Correo electrónico	IMAP	Internet Message Access Protocol	Abierto	4.0	N/A	Estable
		POP3	Post Office Protocol versión 3	Abierto	3.0	N/A	Estable
		SMTP	Simple Mail Transfer Protocol	Abierto	RFC 5321	N/A	Estable



Protocolos de comunicación e intercambio - Tecnologías de transporte y red	AtomPub	Atom Publishing Protocol	Abierto	N/A	.atompub .xml	Estable
	CDN	Content Delivery Network	Abierto	N/A	N/A	Estable
	DNS	Domain Name System	Abierto	RFC 1035	N/A	Estable
	FTP	File Transfer Protocol	Abierto	RFC 2428	N/A	Estable
	FTPS	File Transfer Protocol Secured	Abierto	RFC 4217	N/A	Estable
	FTP over SSH	File Transfer Protocol over Secure Shell	Abierto	N/A	N/A	Estable
	HTTP	HyperText Transfer Protocol	Abierto	1.1 RFC 2616 RFC 2817	N/A	Estable
	ICAP	Internet Content Adaptation Protocol	Abierto	RFC 3238	N/A	Estable
	IPSec	Security Architecture for the Internet Protocol	Abierto	RFC 2401 RFC 4302 RFC 4835	N/A	Estable
	NTP	Network Time Protocol	Abierto	RFC 5905	N/A	Estable
	SFTP	SSH File Transfer Protocol	Abierto	2.0	N/A	Estable



Integración de aplicaciones y servicios	Autenticación - Certificados	OCSP	Online Certificate Status Protocol	Abierto	RFC 2560	N/A	Estable
	Protocolos de comunicación e intercambio - Servicios Web	SOAP	Simple Object Access Protocol	Abierto	1.1	N/A	Estable
		UDDI	Universal Discovery, Description and Integration	Abierto	3.0	N/A	Estable
		WSDL	Web Services Definition Languages	Abierto	1.1	N/A	Estable
		WS-Security	Web Services Security	Abierto	1.1	N/A	Estable
	Tecnologías para identificación	ASN.1	Abstract Syntax Notation One	Abierto	2008	N/A	Estable
		URI	Uniform Resource Identifier	Abierto	RFC 3986 RFC 5785	N/A	Estable
		URL	Uniform Resource Locators	Abierto	RFC 1738	N/A	Estable
		URN	Uniform Resource Names	Abierto	N/A	N/A	Estable
	Semántica	DCAT	Data Catalog Vocabulary	Abierto	N/A	N/A	Estable



Integración de datos	Semántica - Metadatos	ISO 8601	Data elements and interchange formats - Information interchange - Representation of dates and times	Abierto	N/A	N/A	Estable	
		MoReq	Model Requirements for the management of electronic records	Abierto	N/A	N/A	Estable	
		PREMIS	PREservation Meta-data: Implementation Strategies V2.1	Propietario	N/A	N/A	Estable	
	Semántica - Tecnologías semánticas	N3 RDF RDFa	Notation 3		Abierto	N/A	.n3	Estable
			Resource Description Framework		Abierto	1.0	N/A	Estable
			Resource Description Framework - in - attributes		Abierto	2008	N/A	Estable



Tecnologías de integración de datos	ATOM	Atom Syndication Format	Abierto	1.0	.atom .xml	Estable
	JSON	JavaScript Object Notation	Abierto	RFC 7159 and ECMA-404	.json .jsonp	Estable
	JSON-RPC	JavaScript Object Notation - Remote Procedure Call	Abierto	1.0	.json .jsonp	Estable
	ODATA	Open Data Protocol	Abierto	2.0	.odata .xml	Estable
	RSS	Really Simple Syndication	Abierto	2.0	.rss .xml	Estable
	XML	Extensible Markup Language	Abierto	1.0	.xml	Estable
	XSD	XML Schema	Abierto	1.0	.xsd	Estable

Anexo No. 5. Metodología de desarrollo del software gubernamental.





## **EQUIPO DE TRABAJO**

### **Dirección General**

Armando García, Director General

### **Departamento de Estandarización, Normativas y Auditoría Técnica (ENAT)**

Elvyn Peguero, Gerente del ENAT

Shalem Pérez, Auditor de Estándares NORTIC

Winner Núñez, Auditor de Estándares NORTIC

Ginsy Aguilera, Analista de Estándares y Normativas

Hamlet Durán, Analista de Estándares y Normativas

Enyer Pérez, Analista de Estándares y Normativas

Ariel Acosta, Consultor de Estándares y Normativas

### **Comité Interno para Evaluación de las Normas (CIEN) – Equipo OPTIC**

Charli Polanco, Director de TIC

José Luis Liranzo, Director de DIGOB

Miguel Guerra, Gerente Multimedia

### **Comité de Estándares de Tecnologías de la Información y Comunicación (COETIC)**

Dahiri Espinosa

### **Dirección General de Ética e Integridad Gubernamental (DIGEIG)**

Alfonso Espinal

### **Instituto Dominicano de las Telecomunicaciones (INDOTEL)**

Francis Valdez Soto

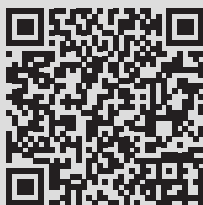
Cámara de Cuentas











Para Visualizar y descargar  
este documento leer este  
código

Av.27 de Febrero #419, Santo Domingo, R.D.  
Tel.:+ 809.286.1009 info@optic.gob.do  
www.optic.gob.do www.dominicana.gob.do



OpticRD