

Lineamientos para la Organización del Desarrollo, Publicación, Documentación y Licenciamiento del software del Estado Plurinacional de Bolivia

**Lineamientos para la Organización del Desarrollo,
Publicación, Documentación y Licenciamiento del
software del Estado Plurinacional de Bolivia**

Lineamientos para la Organización del Desarrollo, Publicación, Documentación y Licenciamiento del software del Estado Plurinacional de Bolivia

DS-001

Este documento ha sido elaborado por los miembros del Consejo para las Tecnologías de Información y Comunicación del Estado Plurinacional de Bolivia (CTIC-EPB).

Coordinación Secretaría Técnica del CTIC-EPB: Khantuta Muruchi y Carolina Ovale

Grupo de Trabajo de Desarrollo de Software: Alberto Inch Sainz, Cecilia Leonor Alvarado Monrroy, Edgar Saïre Aruquipa, Elmer Mendoza, Emilio Carlos Tarifa, Grover Pinto Esquivel, Javier Tito Coriza, Justo I. Quispe Poma, Luis Castro Montes, Luis Landaeta Velásquez, Marcelo Salas Vargas, Neyva Carol Silva Gutiérrez, Omar Gutiérrez Condori, Ramiro Loza Herrera, Renato Apaza, Tito Reynaldo Antonio Aquino Castro, Rodolfo Sirpa Calderon.

Diseño: Orestes Sotomayor

Diagramación: Jorge Dennis Goytia Valdivia

Cuidado de edición: Montserrat Fernández

Deposito Legal: XX-XXXX-XX

Se autoriza la reproducción total o parcial de este documento citando la fuente, así como el uso del mismo para obras derivadas que se distribuyen en las mismas condiciones.

La Paz, Bolivia
2018

ctic) CONSEJO PARA LAS
TECNOLOGÍAS DE INFORMACIÓN
Y COMUNICACIÓN

Contenido

1	Introducción	7
2	Marco normativo referencial	11
3	Objetivo	16
4	Alcance	16
5	Desarrollo comunitario	17
5.1	Comunidad de software	17
5.2	Gobernanza del proyecto de software libre	18
5.2.1	Comité de dirección del proyecto	19
5.2.2	Mecanismos de comunicación.....	20
6	Publicación	21
6.1	Información General del Software (Metadata).....	21
6.2	Código fuente en repositorio	23
7	Documentación	28
7.1	Documentación de administración.....	28
7.2	Documentación de desarrollo.....	29
7.3	Documentación del código	30
8	Licencias de software y contratación	31
9	Glosario	32

1 Introducción

El Consejo para las Tecnologías de Información y Comunicación del Estado Plurinacional de Bolivia (CTIC-EPB) se constituye en una instancia de coordinación técnica para la implementación de Gobierno Electrónico y para el uso y desarrollo de Tecnologías de Información y Comunicación en el país.

Entre las principales tareas asignadas al CTIC-EPB se encuentran:

- Formular propuestas de políticas y normativa relacionada con Gobierno Electrónico, a ser presentadas a la AGETIC;
- Presentar proyectos y programas de Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental a la AGETIC para su gestión;
- Generar mecanismos de participación de instituciones y organizaciones de la sociedad civil en la proposición y formulación de políticas y acciones relacionadas con Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental;
- Establecer espacios de coordinación entre las entidades del sector público para el desarrollo conjunto de programas, proyectos o acciones de Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental;
- Desarrollar y proponer estándares abiertos oficiales del Estado Plurinacional de Bolivia en materia de Gobierno Electrónico y Tecnologías de Información y Comunicación aplicables a las entidades del sector público;
- Establecer espacios de coordinación de comunidades de desarrollo informático, dentro del Estado, con la ciudadanía y a nivel internacional.

El 5 de mayo de 2016 se llevó a cabo la inauguración y la primera reunión del Pleno del CTIC-EPB, en la que se conformaron seis grupos temáticos de traba-

jo: Interoperabilidad, Software Libre, Seguridad, Infraestructura, Desarrollo de Software y Datos Abiertos.

Cada Grupo de Trabajo estuvo integrado por servidoras y servidores públicos de las entidades del nivel central del Estado: Órgano Legislativo, Judicial, Electoral y Ejecutivo, incluyendo sus instituciones descentralizadas, autárquicas, empresas públicas y autoridades de regulación sectorial, Ministerio Público y Procuraduría General del Estado.

Adicionalmente, se invitó a participar, en calidad de miembros adjuntos, a representantes de entidades territoriales autónomas, universidades públicas e indígenas y sociedad civil, a fin de trabajar y elaborar propuestas a ser presentadas al Consejo para su posible implementación a nivel estatal.

Cabe mencionar que el desarrollo de los Grupos de Trabajo y del Consejo se enmarca en el Reglamento de Funcionamiento del CTIC-EPB, aprobado mediante la Resolución Administrativa N° 024/2016 de la AGETIC, de fecha 31 de mayo de 2016.

En cumplimiento del Plan de Implementación de Software Libre y Estándares Abiertos, aprobado Mediante **el Decreto Supremo N° 3251 de 12 de julio de 2017**, la AGETIC como secretaría técnica del CTIC-EPB se puso como objetivo elaborar un documento que defina la Publicación, Documentación y Licenciamiento de software libre en el Estado Plurinacional de Bolivia que sirva como lineamiento para la creación de comunidades de desarrollo de software en el Estado y así promover las condiciones para la generación de conocimiento en pos de la soberanía tecnológica.

Es importante resaltar a las entidades, órganos del Estado y representantes de la sociedad civil, que participaron a través de sugerencias y acotaciones al documento inicial elaborado por el grupo. Entre estas entidades se encuentran:

- Bono Juana Azurduy (BJA)
- Oficina Técnica para el Fortalecimiento de la Empresa Pública (OFEP)

- Yacimientos Petroleros Fiscales Bolivianos (YPFB)
- Fondo Nacional de Desarrollo Regional (FNDR)
- Unidad de Análisis de Políticas Sociales y Económicas (UDAPE)
- Instituto Nacional de Estadística (INE)
- Ministerio de Economía Finanzas Públicas (MEFP)
- Dirección General de Servicio Civil (DGSC)
- Autoridad de Impugnación Tributaria (AIT)
- Autoridad de Fiscalización y Control de Pensiones y Seguros (APS)
- Observatorio Plurinacional de Calidad Educativa (OPCE)
- Ministerio de Educación (MIN-EDU)
- Servicio Nacional de Patrimonio del Estado (SENAPE)
- Instituto Boliviano de Metrología (IBMETRO)
- Banco Central de Bolivia (BCB)
- Agencia para el Desarrollo de la Sociedad de la Información en Bolivia (ADSIB)
- Instituto Geográfico Militar (IGM)
- Ministerio de Justicia y Transparencia Institucional (MIN-JTI)
- Boliviana de Aviación (BOA)
- Servicio Nacional de Patrimonio Intelectual (SENAPI)

- Autoridad de Fiscalización del Juego (AJ)
- Registro Único para la Administración Tributaria (RUAT)
- Aduana Nacional (AN)
- Servicio Nacional de Sanidad Agropecuaria e Inocuidad Alimentaria (SENASAG)
- Instituto Nacional de Reforma Agraria (INRA)
- Autoridad de Fiscalización y Control Social de Electricidad (AE)
- Servicio Nacional de Impuestos (SIN)
- Empresa Estatal de Transporte por Cable "Mi Teleférico"
- Ministerio de Trabajo, Empleo y Previsión Social (MTEPS)
- Instituto Boliviano de Ciencia y Tecnología Nuclear (IBTEN)
- Autoridad de Supervisión del Sistema Financiero (ASFI)
- Agencia de Gobierno Electrónico y Tecnologías de la Información y Comunicación (AGETIC)
- Pedro Carranza (sociedad civil)

2 Marco normativo referencial

El marco normativo concerniente a la temática incluye:

- El Artículo 1° (Objetivo) del Decreto Supremo 24582, de 25 de abril de 1997, Reglamento del Soporte Lógico o Software, de conformidad al inciso "l" artículo 6 de la Ley de Derecho de Autor, de 13 de abril de 1992, el reglamento regula los derechos de los autores y titulares de derechos de autor y define el régimen de protección del soporte lógico y las relaciones de explotación del mismo. El derecho de autor nace con la creación de la obra, de acuerdo a lo previsto en el Art. 2 de la Ley N° 1322.
- El Artículo 2° (Definiciones) del Decreto Supremo 24582, tiene entre algunas de sus definiciones relacionadas al tema, las que se describen a continuación:
 - **Licencia de uso:** Documento mediante el cual se otorga autorización de uso no-exclusivo y no-transferible del soporte lógico o software, de acuerdo a los términos y condiciones mencionados en el presente reglamento. Este convenio de licencia le permite a un solo usuario instalar el software en una sola computadora y un solo lugar y una sola vez, excepto acuerdo tácito en el mismo que amplíe dichas condiciones.
 - **Soporte lógico:** (Software) El soporte lógico es un conjunto de uno o varios programas para computadora, puede incluir información de apoyo, documentación y material auxiliar, cualquiera sea su forma de expresión y fijación.
- El Artículo 12° (Obras por Encargo) del D.S. 24582 señala que "El soporte lógico y el banco de datos que se cree bajo un contrato laboral o de prestación de servicios y/o el que fuera desarrollado por empleados o funcionarios públicos en cumplimiento de las obligaciones inherentes a sus cargos, tendrán como titular a la persona natural o jurídica por cuya cuenta y riesgo se realizan, salvo que exista un convenio o contrato que indique lo contrario de conformidad al artículo 29 de la Ley N° 1322".

- El Artículo 77 (Software libre) de la Ley N° 164, de 8 de agosto de 2011, Ley General de Telecomunicaciones, Tecnologías de Información y Comunicación señala en el párrafo I: “Los Órganos Ejecutivo, Legislativo, Judicial y Electoral en todos sus niveles promoverán y priorizarán la utilización del software libre y estándares abiertos, en el marco de la soberanía y seguridad nacional”.
- El Artículo 3 (Definiciones) del Decreto Supremo N.º 1793, Reglamento para el Desarrollo de Tecnologías de Información y Comunicación, además de las definiciones técnicas establecidas en la Ley N° 164, para el cumplimiento del presente Reglamento, adopta las siguientes definiciones:

1. Software libre: Software licenciado por su autor, bajo una licencia de código fuente abierta, de manera tal que permita al usuario el ejercicio de las siguientes libertades:

1. Ejecutar el software, para cualquier propósito, sin restricción alguna;
2. Estudiar cómo funciona el software y modificarlo para que cumpla un determinado propósito, a través del acceso al código fuente del mismo y todos los componentes que hacen posible su funcionamiento. El acceso al código fuente es una condición necesaria e imprescindible;
3. Redistribuir copias del software;
4. Distribuir copias de las versiones modificadas a terceros. El acceso al código fuente es una condición necesaria e imprescindible.

2. Software propietario o software privativo: Todo software que no cumpla parcial o totalmente con cualquiera de las condiciones mencionadas para el software libre, se considera para los efectos del presente Reglamento, software propietario;

3. Estándar abierto: Es una especificación técnica o protocolo normalizado:

- a. Cuyas especificaciones técnicas, completas y coherentes, están sujetas a una evaluación pública completa, se puede usar sin restricciones y está disponible por igual para todos los usuarios y/o partes, sin costo alguno para su uso;
 - i. Que no necesita ningún componente o extensión adicional que tenga dependencias con formatos o protocolos que no cumplan la definición de Estándar Abierto;
 - ii. Que está libre de cláusulas legales o técnicas que limiten o restrinjan su utilización por cualquier usuario y/o parte o en cualquier modelo de negocio;
 - iii. Que es gestionado y puede ser desarrollado independientemente por cualquier organización en un proceso abierto a la participación equitativa e inclusiva de competidores, usuarios, especialistas del área de aplicación y terceras partes;
 - iv. Que esté disponible en al menos una implementación completa, cuya documentación y especificación técnica está disponible para todas las partes con grado de detalles suficientes para un desarrollo correcto y de calidad.
- La disposición transitoria quinta, del D.S 1793, establece que el plazo máximo para la migración de los sistemas de las entidades públicas a Software Libre y Estándares Abiertos será de siete (7) años desde el inicio de la migración.
- El Artículo 1.- (Objeto) del Decreto Supremo N° 3251 DE 12 de julio de 2017 establece en su inciso b) Aprobar el Plan de Implementación de Software Libre y Estándares Abiertos que en Anexo forma parte integrante del presente Decreto Supremo.
- La disposición transitoria sexta determina que a los cuatro (4) años de aprobado el Decreto Supremo N° 3251, la AGETIC en coordinación con las enti-

dades del sector público realizará la evaluación al proceso de implementación de Software Libre y Estándares Abiertos, para considerar el estado de situación y el nivel de cumplimiento en los aspectos técnicos, financieros, administrativos y normativos y efectuar los ajustes necesarios, producto de la evaluación.

- La disposición final segunda dispone:
 - I. En caso de adquisición o donación, ampliación y/o renovación de Licencias de Software Propietario o desarrollo de aplicaciones en plataforma de Software Propietario de las entidades del sector público, en aplicación a la previsión establecida en el Artículo 23 del Reglamento para el Desarrollo de Tecnologías de Información y Comunicación aprobado mediante Decreto Supremo N° 1793 de 13 de noviembre de 2013, la ADSIB dará a conocer su conformidad u oposición en el marco del reglamento específico aprobado por esta, que contendrá los plazos y procedimientos determinados para la presente previsión.
 - II. En caso de que la ADSIB manifieste su conformidad, esta podrá tener una validez por más de una gestión, pero en ningún caso este periodo podrá superar los plazos establecidos en el Anexo aprobado por el presente Decreto Supremo y en el Decreto Supremo No 1793”.
- El Plan de Implementación de Software Libre y Estándares Abiertos (aprobado mediante Decreto Supremo N° 3251 de 12 de julio de 2017) establece que: “todo sistema o software adquirido o desarrollado por entidades públicas sea a través de procesos de contratación, adquisición, desarrollo interno o cualquier otro, deberá contemplar los estándares a ser adoptados por el Estado Plurinacional de Bolivia en el Catálogo de Estándares Abiertos”.
- Los estándares, entornos y lenguajes, al momento de realizar la concepción, desarrollo, adquisición de nuevos sistemas o software o en el caso de actualizaciones, deben considerar los siguientes aspectos:
 - Estándar abierto de desarrollo y de control de la calidad a ser utilizado

- Lenguaje de programación
 - Librerías y dependencias
 - Base de datos
 - Herramientas de programación, manejo de versiones y compilación
 - Entornos de desarrollo, pruebas y producción: sistema operativo, servicios de base
 - Mecanismos de interoperabilidad: protocolos, servicios web, formatos de entrada, de salida y de almacenamiento interno
- Asimismo, el mencionado plan establece que el CTIC-EPB establecerá los lineamientos para la publicación, documentación y licencia del software desarrollado.

3 Objetivo

El presente documento tiene como objetivo establecer los lineamientos para la organización del desarrollo, publicación, documentación y licenciamiento de software libre en el Estado Plurinacional de Bolivia.

4 Alcance

En este documento se formulan los lineamientos para estandarizar el proceso de organización del desarrollo de un proyecto de software libre en el Estado Plurinacional de Bolivia. Este proceso considera cuatro etapas en la organización del desarrollo de un proyecto de software libre: el desarrollo del software, la publicación del software, la documentación generada y el licenciamiento del software.

En primera instancia, en lo que se refiere a la organización del desarrollo del software se propone un modelo de trabajo vinculado a la filosofía del software libre y los paradigmas de soberanía tecnológica que se ha propuesto el Estado Plurinacional de Bolivia

En segunda instancia, se establecen lineamientos y mecanismos que orientan el proceso de publicación de un proyecto de software libre en el Repositorio Estatal. Este proceso contempla los lineamientos para la generación de la documentación necesaria que acompaña el proyecto de software.

Por otro lado, se formulan los lineamientos acerca del licenciamiento y los aspectos relacionados a los procesos de contratación para el desarrollo de software libre.

Es necesario aclarar que en este documento no se incluyen lineamientos vinculados al proceso de producción del software, es decir a la parte metodológica, la arquitectura y las soluciones técnicas del mismo.

5 Desarrollo comunitario

Se define el desarrollo comunitario como un modelo organizativo que promueve y establece las condiciones técnicas y los mecanismos necesarios para el desarrollo colectivo y colaborativo de software en torno a comunidades de programación, generando un modelo de aprovechamiento de los resultados que beneficia a todos los usuarios (desarrolladores o no) de la misma manera.

El software libre se basa en el desarrollo comunitario. Es decir que, para el desarrollo, actualización, modificación, revisión y mantenimiento de un determinado software se requiere de la participación de varias personas que trabajen de manera mancomunada en estas actividades.

Para que esto sea posible, es necesario que el desarrollo de software, desde un inicio, se vincule en el Repositorio Estatal de Software Libre. De esta manera, se facilitará la participación de las y los servidores públicos, la sociedad civil y todos los actores que deseen contribuir en el desarrollo de un proyecto de software.

A partir de la aprobación del presente lineamiento, la ADSIB, al ser la encargada del Repositorio Estatal, deberá prever un procedimiento para el desarrollo comunitario de software en el Repositorio Estatal de Software Libre, considerando y preservando los derechos de registro, titularidad y autoría desde el primer día de desarrollo, en el marco de la normativa vigente y la Licencia Pública General (LPG-Bolivia).

Es recomendable que para el desarrollo de software libre del Estado Plurinacional de Bolivia cada proyecto deba tener una comunidad de software y una gobernanza del proyecto.

5.1 Comunidad de software

La comunidad de software está definida como aquellos miembros que participan y contribuyen en el desarrollo colectivo, comunitario y colaborativo de un proyecto de software. La contribución de cada uno de estos miembros beneficia a todos los involucrados de la comunidad.

Esta forma de organización comunitaria en el desarrollo de software ofrece la posibilidad para que cada miembro pueda revisar, mejorar y proponer funcionalidades en el proyecto de software. Dado que el código fuente se encuentra abierto y disponible a los miembros de la comunidad, maximiza el potencial por la cantidad de personas dispuestas a inspeccionar, aportar y sugerir mejoras o correcciones de un determinado proyecto de software.

La comunidad de usuarios puede estructurarse según su tamaño. Existen comunidades pequeñas compuestas por dos o tres miembros; las comunidades medianas y las comunidades grandes, compuestas por una multitud de colaboradores.

En este sentido, para la conformación de una comunidad de software en el Estado Plurinacional de Bolivia se deberá tomar en cuenta los siguientes lineamientos:

- Se debe generar una comunidad de software libre mínimamente con 2 miembros.
- La comunidad podrá estar conformada por servidoras y servidores públicos, miembros de la comunidad de desarrollo de software libre y sociedad civil.
- Para formar parte de una comunidad no es necesario tener conocimientos técnicos, ya que se puede aportar con el conocimiento sobre el área de acción del software, con pruebas, traducciones y aportes a la documentación.
- El tamaño de la comunidad no deberá ser una limitante para coordinar un proyecto.

5.2 Gobernanza del proyecto de software libre

Se define la gobernanza de un proyecto de software libre como un modelo organizativo vinculado a la coordinación del desarrollo de software y a los mecanismos de comunicación que coadyuvan en la dirección del proyecto. Para la coordinación del desarrollo de software libre se realizará la conformación de un Comité de dirección del proyecto.

5.2.1 Comité de dirección del proyecto

Conformación y funciones del Comité de Dirección

Se conformará el Comité de dirección encargado de realizar las tareas de coordinación del proyecto de software libre de acuerdo al tamaño y alcance de la entidad pública.

Este comité estará conformado por algunos miembros de la comunidad del proyecto de software, bajo los siguientes criterios:

- La elección de miembros estará liderada por la entidad titular del proyecto o la entidad encargada del Repositorio de Software Libre.
- La elección de miembros puede ser por meritocracia, institucionalidad, por elección de la comunidad o cualquier otro modelo que se defina en el proyecto.

Las funciones y responsabilidades del Comité de dirección del proyecto serán las siguientes:

- Organizar los alcances y objetivos del proyecto
- Ordenar las tareas de desarrollo, recibir todos los aportes, analizarlos y aprobarlos.
- Definir las políticas y estrategias para el desarrollo del software
- Aceptar o no los aportes realizados por la comunidad.
- Definir cuándo el proyecto deja de estar activo.

5.2.2 Mecanismos de comunicación

Todo proyecto de desarrollo de software debe establecer mecanismos y canales de comunicación para garantizar que toda la información y documentación necesaria esté al alcance de los participantes y del público en general. Inicialmente, se debe partir del principio de que todos los elementos o componentes de un proyecto de software tienen que estar publicados.

La coordinación a través de una comunidad requiere establecer un canal de coordinación para el desarrollo de software. Para esto, se debe generar un espacio en la web que pueda reflejar la hoja de ruta de desarrollo de software.

En este sentido, los contenidos del proyecto de software que deben publicarse en el sitio web son:

1. Objetivo y alcance del proyecto
2. Reglas para el desarrollo del proyecto
3. Reglas para la contribución al proyecto
4. Reglas de la comunidad alrededor del software
5. Enlaces al repositorio de código, si el sitio web no estuviera en el mismo repositorio
6. Listado de funcionalidades, hitos, fechas y versiones del software
7. Herramientas de comunicación (listas de correo, email, chat, etc.)
8. Canales de reporte de errores

Los lineamientos de la implementación de los mecanismos y canales de comunicación serán especificados en otros documentos del CTIC-EPB.

6 Publicación

El Decreto Supremo 1793 del Reglamento para el Desarrollo de Tecnologías de la Información y Comunicación establece la obligatoriedad de publicar el software desarrollado por o para las entidades públicas del Estado Plurinacional de Bolivia en el Repositorio Estatal de Software Libre, con excepción de aquellas consideradas estratégicas por cada institución.

La publicación de un proyecto de software libre en el Repositorio Estatal fomenta:

- a. la generación de comunidad en torno a un proyecto;
- b. compartir conocimiento;
- c. maximizar el impacto para el beneficio de la población.

De esta manera, el software puede evolucionar en su funcionalidad y su alcance. Estos esfuerzos conjuntos pueden abaratar costos, dejando a la entidad a cargo del proyecto la responsabilidad de difundir sus proyectos y hacerlos visibles ante la sociedad.

Para la publicación de un proyecto de software libre en el repositorio Estatal se debe contar con una información mínima, la misma que será detalla a continuación.

6.1 Información General del Software (Metadata)

- a. **Los colaboradores:** Todos los que intervinieron en el desarrollo del software tienen que estar registrados como colaboradores. Este registro podrá ser verificado con la lista de usuarios que realizaron aportes al repositorio o que participaron en el desarrollo y planificación en las instituciones. El mismo puede realizarse mediante un seudónimo o dirección de correo.

b. La titularidad: La titularidad corresponde a la entidad que inicia el desarrollo del proyecto. Esta entidad entrega a la comunidad el software bajo los términos de la licencia LPG-Bolivia.

Si dos o más entidades iniciaran el desarrollo al mismo tiempo, las mismas tendrán que definir a una de ellas para establecer la titularidad.

Además, el Repositorio Estatal, exige la siguiente información:

Dato	Descripción
Datos Generales	
Objetivo y alcance del software	Detallar información sobre el software, el objetivo, su uso y alcance
Fecha de inicio de desarrollo	La fecha que se dio inicio al proyecto
Versión	Versión actual de desarrollo de software
URL de la Comunidad	Información relacionada a la comunidad que tiene a su cargo el mantenimiento, desarrollo y continua mejora del software. Página web, donde se pueda encontrar información sobre el desarrollo
URL del Repositorio de Código	Incluir la dirección donde se trabaja el código fuente, puede ser en una forja de la institución o del repositorio Estatal
Especificaciones técnicas	
Sistemas operativos	Sistemas operativos en los que puede ser instalado el software
Lenguajes de programación	Detalle de los lenguajes de programación utilizados para el desarrollo del software
Manejador de Base de Datos	Detalle del manejador de base de datos que utiliza el software
Dependencias	Detalle de las dependencias del software
Se deberá incluir:	
<ul style="list-style-type: none"> ▪ Manuales existentes (uso, instalación, administración, etc.) ▪ La licencia de software libre del proyecto 	

La ADSIB podría adicionar o prescindir de algunos requerimientos listados en este punto, de acuerdo a la conveniencia del procedimiento de publicación de software libre en el Repositorio Estatal.

6.2 Código fuente en repositorio

El código fuente del software tiene que estar publicado en el repositorio Estatal de Software Libre del Estado Plurinacional de Bolivia, de acuerdo a los siguientes criterios:

- Debe utilizarse un sistema de manejo de versiones y seguimiento de cambios compatible al repositorio mencionado.
- Dicho sistema debe permitir centralizar colaboraciones, reportes de errores, nuevas funcionalidades y solicitud de información.
- El repositorio deberá estar actualizado con la versión más reciente del proyecto, siendo esta una responsabilidad del titular del software desarrollado, mientras el proyecto se encuentra activo.
- Podrán ser adicionados al repositorio del proyecto de software todos los usuarios que deseen brindar aportes.
- Para mantener un orden lógico, se recomienda crear un grupo bajo el nombre de la entidad titular del proyecto donde se incluirá el proyecto de desarrollo de software.
- Para que se puedan manejar los errores, las solicitudes de nuevas funcionalidades o cualquier reporte tienen que utilizar las herramientas del repositorio de software estatal.
- También se deben utilizar las mismas para coordinar y mantener las versiones relacionadas a cambios o mejoras.

a. Manejo de versiones

Para establecer el nivel de desarrollo de un proyecto de software es necesario realizar un manejo de versiones, que comprende la asignación de un nombre, có-

digo o número único al conjunto de código fuente realizado hasta un momento determinado.

Los criterios para realizar un manejo de versiones de un proyecto de software libre son:

- El uso de un código único como identificador de las versiones.
- Dicho código será el resultado de la combinación de la sigla o nombre del proyecto y la numeración X.Y.Z.

[Nombre Proyecto] v[X . Y . Z]

Donde:

X : se refiere a los grandes cambios y mejoras que se realiza en el software, y/o crea una incompatibilidad con la versión anterior.

Y : se refiere a las nuevas funcionalidades que son incorporadas al software, realizando cambios menores.

Z : se refiere a la corrección de errores al software.

Información sensible: Si existiese información en los archivos a ser versionados que sea confidencial, sensible o especifique configuraciones inherentes al entorno de trabajo, como ser direcciones IP, puertos, usuarios, contraseñas y otros esta información deberá ser reemplazada por datos de muestra.

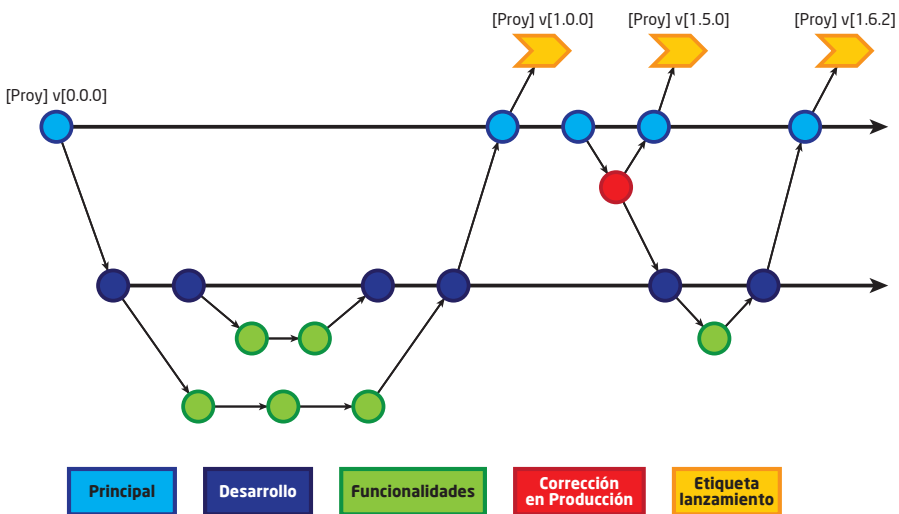
b. Manejo de Ramas

La gestión de un código de un proyecto de software se debe realizar a través de la organización de ramas de desarrollo. Las ramas son hilos de trabajo que permiten avanzar en el desarrollo del código de manera paralela e independiente, logrando obtener un resultado único al final de cada rama.

En el desarrollo de un proyecto de software debe tomarse en cuenta al menos las siguientes ramas: a. Rama principal; b. Rama de desarrollo; c. Ramas de funcionalidades y d. Correcciones en producción. Asimismo, en este proceso es importante considerar la creación de etiquetas de lanzamiento, las cuales marcan los hitos de desarrollo del software y son necesarias para la puesta en producción.

Por otro lado, las ramas antes mencionadas podrán ser asignadas con un nombre convencional (por ejemplo, Principal, Desarrollo, etc.) según el criterio de los desarrolladores y la tecnología que se esté utilizando para la gestión de ramas, siempre y cuando mantenga la descripción de cada rama.

Se sugiere el siguiente esquema para el desarrollo de código por ramas:



- **Rama Principal:** Es la rama central del proyecto y contiene las versiones candidatas para la puesta en producción de un proyecto de software. Se sugieren dos acciones para trabajar en esta rama:
 1. Actualización de archivos: la unión (*merge*) de cambios hacia esta rama debe ser autorizada por el administrador del proyecto y debe estar sujeta a pruebas ejecutadas por el equipo de desarrollo del proyecto.

2. Generación de etiquetas de lanzamiento: respecto a la etapa de puesta en producción del contenido de esta rama se debe cumplir con el siguiente requisito:
 - El equipo asignado debe haber concluido satisfactoriamente las pruebas realizadas antes de generar la etiqueta.
- **Rama de desarrollo:** Es la rama de la versión estable en desarrollo; contiene la última versión del proyecto de software y cuenta con 2 acciones para trabajar en la misma:
 1. Unión (*merge*) con la rama principal: se realiza mediante una solicitud de unión; de esa manera, el paso de cambios a la rama principal será autorizado por el administrador del proyecto después de que haya realizado la verificación mediante las pruebas correspondientes.
 2. Actualización de archivos: todas las versiones que se unen con esta rama deben haber terminado satisfactoriamente las pruebas de los encargados del desarrollo de la versión.
 - **Ramas de Funcionalidades:** Son ramas creadas con la finalidad de desarrollar una característica o funcionalidad del proyecto, pueden estar asociadas a requisitos del proyecto o historias de usuarios para diferenciarlas de las ramas principal y de desarrollo.

Ejemplos:

entregable/servicio-web-v01

entregable/registro-usuarios

entregable/busqueda-paginacion

Estas ramas cuentan con 2 acciones:

1. Creación de la rama, las ramas creadas deben obtenerse de la última versión de la rama de desarrollo.
2. Unión (*merge*) con la rama de desarrollo; para el paso de cambios hacia la rama de desarrollo se deben ejecutar pruebas unitarias y terminarlas satisfactoriamente.

Es importante que estas ramas no sean extensas en tiempo, con la finalidad de evitar conflictos durante la unión (*merge*) con la rama Desarrollo.

- **Correcciones en producción:** Esta rama es de uso excepcional; su finalidad es corregir posibles errores en producción, los cuales requieren soluciones inmediatas. Esta rama debe ser obtenida de la rama principal y una vez solucionado el error debe unirse a la misma y a la rama de desarrollo.
- **Etiqueta de lanzamiento:** Para la generación de estas etiquetas, se ejecutan pruebas sobre la versión de archivos de la rama principal; esta versión de archivos es sometida a pruebas unitarias, de integración, de funcionalidades, además de revisión de calidad del código; si todas las pruebas concluyen satisfactoriamente, se procede con la creación de la etiqueta, la misma es candidata para el paso a producción. Cuando se genera una rama de tipo correcciones en producción y esta se une con la rama principal, se debe seguir el procedimiento de creación de la etiqueta nuevamente.

7 Documentación

La documentación es una actividad esencial en un proyecto de software libre dado que permite compilar información relevante para que otros actores se involucren en el desarrollo del software. Todo proyecto de software debe contar con una estructura documental estándar. La información que esté publicada debe estar organizada sistemáticamente y ser entendible.

Existen dos tipos de categorías para organizar la documentación:

- a. Documentación de administración
- b. Documentación de desarrollo

7.1 Documentación de administración

Este tipo de documentación contempla aquella información relacionada específicamente con la administración del proyecto; es decir, contiene las instrucciones del cómo desplegar el proyecto en los diferentes ambientes comúnmente usados, como el de pruebas, el de desarrollo y el de producción. Así, también se refiere a la documentación que aborda temas de configuraciones y actualizaciones.

Los documentos mínimos que deben incluirse son:

README.md: Este documento describe la solución desarrollada, incluyendo la información de los módulos con los que cuenta, así como la lista de requerimientos que atiende el proyecto y ejemplos de uso del software.

INSTALL.md: Este documento describe los pasos y comandos detallados para la instalación y configuración del software.

UPGRADE.md: Este documento describe los pasos y comandos detallados para la actualización del software.

Los nombres de estos archivos deben estar descritos en inglés para mantener una nomenclatura internacionalmente utilizada que permita la compatibilidad con proyectos internacionales y, de esta manera, promover la colaboración comunitaria.

Estos documentos deben estar elaborados en un formato que pueda ser desplegado en la web, como Markdown (.md) y con el mayor detalle posible en cuanto a casos especiales, complementos, librerías, servicios de interoperabilidad y otros que sean necesarios, para que el proyecto pueda ser desplegado con éxito. Es recomendable que estos documentos se hallen en el directorio raíz del proyecto.

7.2 Documentación de desarrollo

La documentación de desarrollo es aquella que se genera desde el diseño de la solución hasta la culminación del proyecto. Dentro de esta categoría se deberá contemplar la información relacionada a la ingeniería de requerimientos, a la arquitectura y al código generado durante el desarrollo.

- Documentación de la Ingeniería de Requerimientos: se refiere a toda la información generada producto de reuniones realizadas con las partes interesadas, que incluye la recopilación de procesos y requisitos del proyecto de software y todas aquellas tareas que hayan determinado las necesidades o las condiciones a satisfacer por el software a desarrollarse.
- Documentación de la Arquitectura del Software: se refiere a todos los criterios técnicos utilizados en el desarrollo, así como la metodología, el modelo de desarrollo, estructura de directorios, etc.

La documentación mínima sugerida para esta categoría es:

- Modelo de datos: documentación que contiene el esquema de base de datos que utiliza el proyecto.

7.3 Documentación del código

Todo el código desarrollado deberá ser comentado durante su proceso para que las personas que se involucren en el proyecto puedan entenderlo.

Se deben seguir los siguientes lineamientos para la documentación del código:

- Generar la documentación del código de manera automática a través de una herramienta.
- Aplicar la generación de documentación a los siguientes ámbitos: los segmentos de código crítico del proyecto, las funcionalidades y la API de servicios web (si aplica).
- Guardar la documentación generada en un directorio denominado “docs”, ubicado en el directorio raíz del proyecto.
- Generar la documentación en un formato que pueda ser desplegado en la web.

Los manuales de usuario, que incluyen información de los procesos de difusión, capacitación y comunicación, no están contemplados en esta fase debido a que la misma será desarrollada en otra etapa del proyecto por las áreas responsables (área de comunicación, área de recursos humanos, etc.).

8 Licencias de software y contratación

Todo software publicado o desarrollado en el Repositorio Estatal de Software Libre por o para el Estado Plurinacional de Bolivia deberá estar licenciado en favor de la entidad titular bajo la licencia de software libre LPG-Bolivia. Este software no debe tener ningún tipo de dependencia hacia software privativo o propietario (librería, servicio, etc.) para ser licenciado y publicado.

Las dependencias como librerías o servicios deben contar con licencias de software libre tal como se especifica en el Decreto Supremo N° 1793 y, al mismo tiempo, estas no deben tener dependencias de software privativo o propietario de ningún tipo. Debe especificarse la licencia y sus características, si fueran necesarias, de cada librería usada en el proyecto. De esta manera, se evitará incluir en el software otras licencias que puedan anular las libertades con las que ha nacido el proyecto.

El software tiene que estar desarrollado y pensado para ser ejecutado en entornos libres, con librerías y dependencia de servicios que mantengan la misma filosofía del software libre. De la misma manera, el software desarrollado tiene que ser compatible con el hardware que soporta el software libre.

Las licencias de software libre adoptadas por el Estado y sus actualizaciones estarán disponibles en el Repositorio Estatal de Software Libre.

El software libre puede tener más de una licencia, por ejemplo, la licencia LPG-Bolivia y la GPLv3, siempre y cuando tenga como mínimo la licencia LPG-Bolivia y la otra licencia sea compatible con esta.

Respecto a la contratación, todo producto realizado como trabajo de consultoría, al ser propiedad de la entidad titular, podrá ser licenciado en favor de la entidad contratante bajo una licencia de software libre (LPG-Bolivia). Además, esta entidad deberá gestionar la publicación del software en el Repositorio Estatal de Software Libre del Estado Plurinacional de Bolivia. Este registro debe contemplar los requerimientos mencionados en el punto 6.

9 Glosario

Algoritmo: Conjunto predeterminado de instrucciones para resolver un problema específico en un número finito de pasos (compárese con Heurística). (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Banco de datos: Conjunto organizado de información accesible por computadora. (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Heurística: Método de ensayo y error que se vale de reglas empíricas para encontrar la solución para un problema, evaluando por etapas los progresos hechos a lo largo de su curso. (Compárese con Algoritmo). (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Lenguaje de programación: Conjunto de instrucciones con semántica y sintaxis con los cuales se puede desarrollar un programa fuente. (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Markdown: Es un formato de texto plano para escribir documentos estructurados, basado en convenciones de formato de correo electrónico y redes de usuarios. (<https://commonmark.org/>).

Memoria: Dispositivo capaz de recibir datos, retenerlos y suministrarlos a requerimiento del usuario. (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Programa fuente: Conjunto de instrucciones para ser usadas, directa o indirectamente en una computadora a fin de obtener un resultado determinado, en el lenguaje comprensible para el ser humano.

Soporte informático: Todo dispositivo o medio físico (memoria, disquetes, discos duros, cintas, etc.) o medio magnético, óptico, químico o papel y otros empleado para propósitos de comunicación entre humanos y máquinas y fines de almacenamiento. (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

Soporte físico (Hardware): Comprende la totalidad de dispositivos mecánicos, magnéticos, eléctricos y electrónicos en una instalación de procesamiento de datos. (Decreto Supremo 24582 Reglamento del Soporte Lógico o Software, 25 abril de 1997- Artículo 2°.- (Definiciones)).

