



## PLAN DE TRABAJO

# CONTENIDO

<b>CONTENIDO.....</b>	<b>2</b>
<b>GLOSARIO.....</b>	<b>4</b>
<b>INTRODUCCIÓN.....</b>	<b>6</b>
<b>VISIÓN GENERAL DEL PROYECTO .....</b>	<b>7</b>
<b>PLAN DE TRABAJO .....</b>	<b>8</b>
Plan del ciclo de vida del desarrollo .....	10
Calendario y características del proyecto .....	11
<b>ORGANIZACIÓN DEL PROYECTO .....</b>	<b>12</b>
<b>MODELOS DE PROCESOS .....</b>	<b>12</b>
Mapa a nivel de paquetes de trabajo .....	13
<b>PROCESOS DE GESTIÓN.....</b>	<b>14</b>
<b>SUPOSICIONES, DEPENDENCIAS Y RESTRICCIONES .....</b>	<b>14</b>
<b>PROCESO TÉCNICO.....</b>	<b>15</b>
<b>METODOLOGÍAS, TÉCNICAS Y HERRAMIENTAS .....</b>	<b>15</b>
Componentes.....	15
Normatividad aplicable .....	16
<b>PLAN DE MANTENIMIENTO .....</b>	<b>17</b>
<b>PROCESO DE IMPLEMENTACIÓN .....</b>	<b>18</b>
<b>ANÁLISIS DE PROBLEMAS Y MODIFICACIONES .....</b>	<b>19</b>
<b>IMPLEMENTACIÓN DE MODIFICACIONES .....</b>	<b>19</b>
<b>REVISIÓN/ACEPTACIÓN DEL MANTENIMIENTO.....</b>	<b>20</b>
<b>MIGRACIÓN.....</b>	<b>20</b>
<b>PROCESO DE MANTENIMIENTO .....</b>	<b>21</b>



<b>NIVEL DE SOPORTE TECNOLÓGICO .....</b>	<b>22</b>
<b>INCIDENTE.....</b>	<b>22</b>
<b>PROCESO DE NOTIFICACIÓN .....</b>	<b>22</b>
<b>MANTENIMIENTO O CORRECCIONES.....</b>	<b>22</b>

## GLOSARIO

<b>Base de datos</b>	Conjunto de datos pertenecientes a un mismo contexto, almacenados sistemáticamente para su posterior uso.
<b>Batch</b>	Proceso que se realiza de forma secuencial y automática por el ordenador
<b>Código abierto</b>	Permite el acceso al código de programación.
<b>Componentes</b>	Representa el conjunto físico de diversos elementos lógicos, como colaboraciones, interfaces y clases.
<b>Frameworks</b>	Estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos concretos de software.
<b>Frontend</b>	Parte de un sistema de software que interactúa directamente con el usuario
<b>Hardware</b>	Partes físicas de un sistema informático
<b>HTTPS</b>	Protocolo de Transferencia de Hipertexto Seguro
<b>IEEE</b>	Instituto de Ingeniería Eléctrica y Electrónica
<b>JSON</b>	Notación de Objeto Javascript. Formato ligero de intercambio de información
<b>Librerías de desarrollo</b>	Conjunto de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.
<b>Modularización</b>	Programación orientada a objetos. Permite subdividir una aplicación en módulos.
<b>MVC</b>	Patrón de diseño de Modelo-Vista-Controlador (MVC) especifica que una aplicación consta de un modelo de datos, de mecanismos de control e información de presentación.
<b>Oauth2</b>	Protocolo de autorización que permite a terceros acceder a contenidos propiedad de un usuario

<b>PGPS</b>	Plan de Gestión de Proyectos de Software
<b>PMXC</b>	Proyecto México Conectado
<b>Product owner</b>	Define los requerimientos y objetivos del producto o proyecto
<b>SCRUM master</b>	Figura necesaria para la gestión de los recursos, así como la planificación del proyecto
<b>SCRUM</b>	Metodología para el desarrollo de software, se utiliza para disminuir los riesgos en el transcurso del desarrollo del proyecto
<b>Servicios REST</b>	Estilo de arquitectura para desarrollar servicios
<b>Software</b>	Conjunto de programas que permiten desarrollar determinadas tareas
<b>Sprint</b>	Ciclos breves para el desarrollo que proporcionan un resultado completo
<b>TIC</b>	Tecnologías de la Información y la Comunicación

# INTRODUCCIÓN

El programa Apoyo a Mujeres Jefas de Familia implementado por el Gobierno de Jalisco, a través de la Secretaría de Desarrollo e Integración Social, tiene como objetivo el mejorar los ingresos de los hogares de jefatura femenina con dependientes menores de edad, en condiciones de vulnerabilidad por ingresos, monoparentales, que habiten en los 125 municipios de Jalisco, con el fin de contribuir a aumentar sus recursos disponibles para la puesta en marcha de estrategias de sobrevivencia contra la pobreza.

Retos Jalisco es un programa de la Secretaría de Planeación, Administración y Finanzas del Gobierno del Estado de Jalisco que tiene como finalidad brindar nuevas y mejores oportunidades para nuevos emprendedores en Tecnologías de la Información en el Gobierno de Jalisco.

Con este reto se busca el desarrollo de una herramienta que permita establecer un mejor control y seguimiento de cada una de las beneficiarias del programa Apoyo a Mujeres Jefas de Familia, con el propósito de saber cuál ha sido su comportamiento dentro del mismo. La solución debe fungir como una comunidad de las mujeres beneficiadas y las instancias gubernamentales relacionadas, con el fin de crear y sostener una comunicación permanente, además debe permitir acceso al público a la información básica del programa.

El Plan de trabajo, define los procesos técnicos y de gestión necesarios para el desarrollo del sistema. En este documento se plasma la visión general del proyecto y la forma en que éste evoluciona, además se presenta la manera en que es organizado, lo cual induce definir el modelo de los procesos que se ejecutan para integrar el sistema.

El proceso técnico es parte fundamental del plan, debido a que se describen las metodologías, técnicas y herramientas que serán utilizadas, por último se precisa la normatividad que es utilizada para la documentación del desarrollo del software.

## VISIÓN GENERAL DEL PROYECTO

---

La visión general del proyecto consiste en desarrollar un sistema integral para la generación y visualización de reportes unificados, para la consecución de los indicadores, metas y objetivos del programa, cumpliendo con los siguientes objetivos:

- Administrar la información de los programas y las beneficiarias pertenecientes a ellos.
- Explorar los datos con la finalidad de realizar modelos estadísticos que permitan su conversión en información útil para la toma de decisiones operativas y estratégicas, a corto, mediano y largo plazo.
- Informar al público en general sobre los bienes producidos por el programa y las beneficiarias actuales inscritas en ellos
- Brindar una herramienta de consulta y envío de información a las beneficiarias pertenecientes y candidatas.

El propósito del Plan de Desarrollo de Software o Plan de trabajo es proveer información necesaria para llevar un correcto seguimiento del proyecto.

El Plan de Desarrollo de Software considera para la integración del sistema el desarrollo de los siguientes módulos: general, beneficiaria y administrador, los cuales pueden ser consultados en el apartado “Evolución del Plan del Proyecto”.

A continuación se muestra el ciclo de desarrollo de software, el cual se compone de cinco fases:

- Fase 1: Levantamiento de requerimientos
- Fase 2: Análisis y diseño
- Fase 3: Desarrollo
- Fase 4: Pruebas
- Fase 5: Implementación

# PLAN DE TRABAJO

A continuación se presenta el esquema de arquitectura de componentes macro a desarrollar en el sistema:

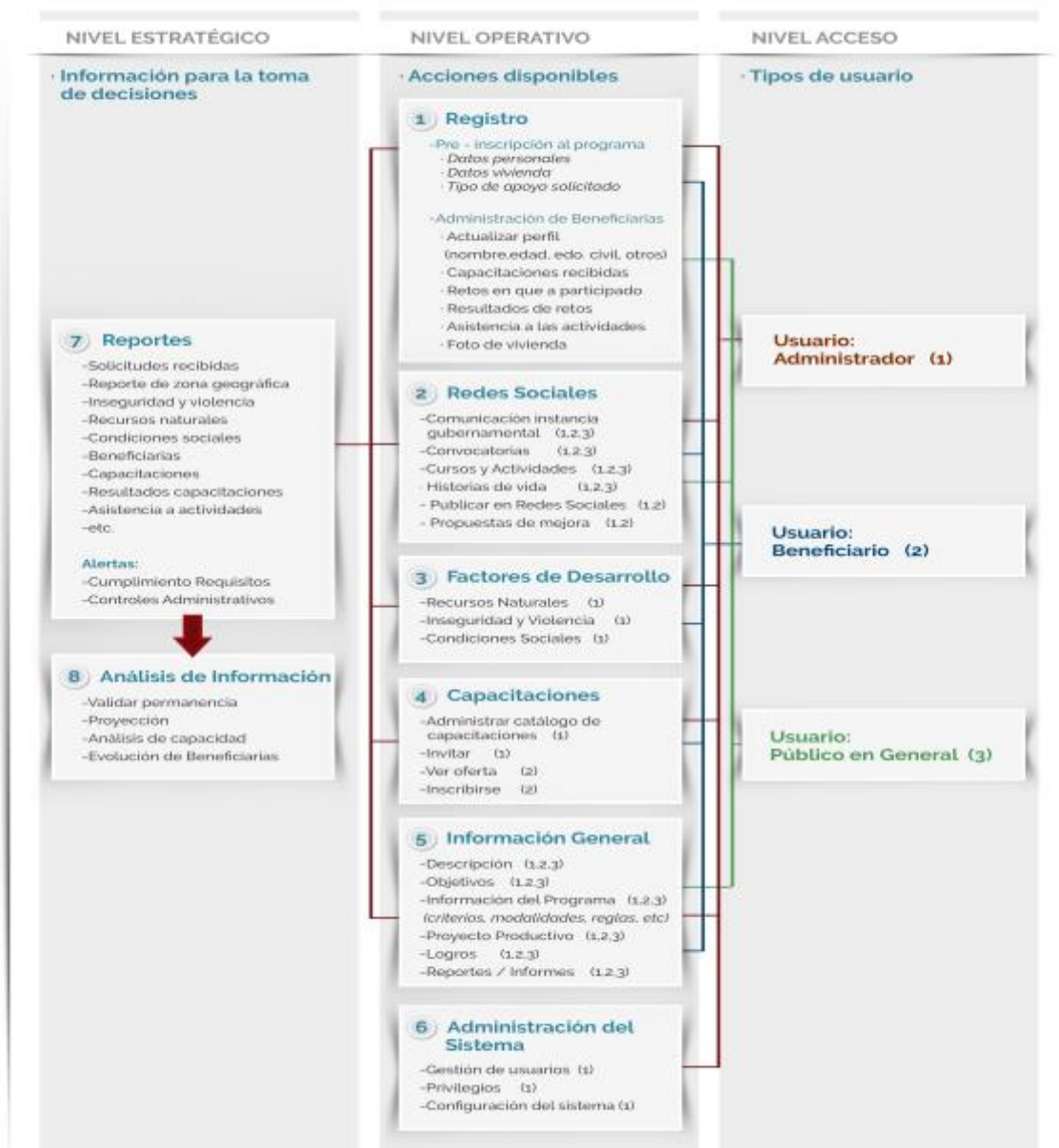


Imagen 1. Arquitectura del sistema. Fuente Elaboración propia



Los módulos tienen diferentes interacciones entre ellos para lograr los objetivos del proyecto. Para el desarrollo los módulos observados previamente se planean realizar cinco fases que se describen a continuación:

### **Fase 1: Levantamiento de requerimientos**

Se abarcan los aspectos a tomar en cuenta para la especificación de requisitos de software, la descripción de los usuarios, disponibilidad y mantenimiento, entre otros. Esta práctica proporciona los insumos necesarios para definir las actividades diarias de cada persona involucrada en el proyecto.

La importancia de la especificación de requisitos es que aporta al equipo de desarrollo una visión general de la expectativa que el cliente tiene del desarrollo del sistema.

### **Fase 2: Análisis y diseño**

En la fase de diseño se establecen las estructuras de datos, arquitectura general del software, representaciones de interfaz y algoritmos, el proceso de diseño traduce requisitos en una representación de software. Además se aborda el modelado integral del software, teniendo presente que no es un producto estático, sino un producto que evoluciona desde su concepción y afecta a las organizaciones transversalmente de una u otra manera.

### **Fase 3: Desarrollo**

En esta fase se define la estructura de la base de datos, se implementan las funciones del sistema, se desarrollan las interfaces y se traduce el diseño del software a programación, al generar estas actividades se obtiene el código.

### **Fase 4: Pruebas**

En el desarrollo del proyecto de software es de vital importancia realizar pruebas para detectar errores o comportamientos no deseados con el propósito de corregirlos antes de entregar el producto final. En esta fase se realiza el testeo de todas las funcionalidades de acuerdo a los sprints definidos y según el protocolo estipulado.

### **Fase 5: Implementación**

Se asegura que el sistema sea operacional o que funcione de acuerdo a los requerimientos solicitados. Es aquí donde se realiza la validación de funcionalidades, se elaboran y reproducen -si es el caso- los

manuales digitales y de acuerdo a la metodología SCRUM, el Product Owner realiza la validación de los módulos.

## Plan del ciclo de vida del desarrollo

Como se aprecia en el apartado anterior el desarrollo del sistema consiste en cinco fases, las cuales comprenden hitos específicos y son descritos en la tabla que se presenta a continuación:

Fase	Hitos
1. Levantamiento de requerimientos	<ul style="list-style-type: none"> <li>• Revisión de documentos de referencia</li> <li>• Revisión de sistemas de referencia</li> <li>• Reuniones de trabajo</li> </ul>
2. Análisis y diseño	<ul style="list-style-type: none"> <li>• Diseño del sistema</li> <li>• Diseño y desarrollo de interfaces visuales</li> <li>• Documentación de desarrollo del sistema</li> <li>• Análisis y diseño por módulo</li> </ul>
3. Desarrollo	<ul style="list-style-type: none"> <li>• Sprint 1 al 8</li> </ul>
4. Pruebas	<ul style="list-style-type: none"> <li>• Validación de funcionalidades</li> <li>• Corrección de errores y bugs</li> <li>• Implementación de mejoras</li> <li>• Sprint 1 al 8 (paralelas)</li> </ul>
5. Implementación y mantenimiento	<ul style="list-style-type: none"> <li>• Elaboración y reproducción de manuales</li> <li>• Definición de product owner y usuarios del sistema</li> <li>• Validación de módulos por product owner</li> <li>• Armado de paquetes de documentación</li> </ul>

**Tabla 1.** Plan de fases. Fuente: Elaboración PROPIA

## Calendario y características del proyecto

A continuación se presenta el calendario del proyecto con las principales tareas y actividades programadas así como los días comprendidos para cada uno de los Sprints.

Las características requeridas por parte del cliente para el desarrollo del sistema se describen a continuación:

Actividad	Duración	Comienzo / Fin
<b>Fase 1: Levantamiento de requerimientos</b>	10 días	1/11/16 a 14/11/16
<b>Fase 2: Análisis y diseño</b>	30 días	17/11/16 al 28/12/16
<b>Fase 3: Desarrollo</b>	40 días	29/12/16 al 21/02/17
<b>Fase 4: Pruebas</b>	40 días	29/12/16 al 21/02/17
<b>Fase 5: Implementación y mantenimiento</b>	10 días	22/02/17 al 06/03/17

**Tabla 2.** Calendario y características del proyecto. Fuente: Elaboración PROPIA

### ■ Usuarios del Sistema

El sistema contiene información de carácter tanto informativo como administrativo. Por ello es vital definir diferentes tipos de usuario con los respectivos privilegios para permitir o restringir determinadas operaciones sobre la información que se maneja.

Los usuarios definidos para los módulos administrador, beneficiarias y general, se enlistan a continuación:

- Administrador
- Beneficiaria
- Usuario público

# ORGANIZACIÓN DEL PROYECTO

## MODELOS DE PROCESOS

---

Para desarrollar el sistema es conveniente llevar a cabo una adecuada gestión de proyectos, por lo que se considera necesario establecer los actores que intervienen. Uno de los actores que tiene mayor preponderancia es el Product Owner (cliente), ya que es el encargado de definir los requerimientos y objetivos del sistema. Una vez que se han definido dichos objetivos, otro de los actores, el SCRUM Master (líder del proyecto), se encuentra en condiciones de elaborar la planeación estratégica para la integración del sistema, es en esta etapa dónde se priorizan las actividades, métodos y técnicas que se utilizan en el desarrollo, así como, mantiene al equipo del proyecto enfocado en los objetivos, gestiona riesgos y establece prácticas que aseguran la integridad y calidad de los artefactos del proyecto.

El control interno que se realiza por parte de Grupo Sitti de las tareas a desarrollar, es a nivel de paquetes de trabajo, la relación entre los mecanismos para dirigir el proyecto y las funciones de soporte son trazadas en este nivel.

El levantamiento de requerimientos es la primera fase que se ejecuta, es aquí donde se abordan temas administrativos y se programan todas y cada una de las reuniones de trabajo necesarias para delimitar los requerimientos, realizar la presentación de las características y funcionalidades del sistema, la revisión de avances y programación de actividades.

La segunda fase tiene relación con el análisis y diseño del sistema para cumplir las expectativas del cliente, se realiza el mapeo de procesos y flujos, así como, se proporcionan características de diseño que el equipo de desarrolladores incluye en el sistema.

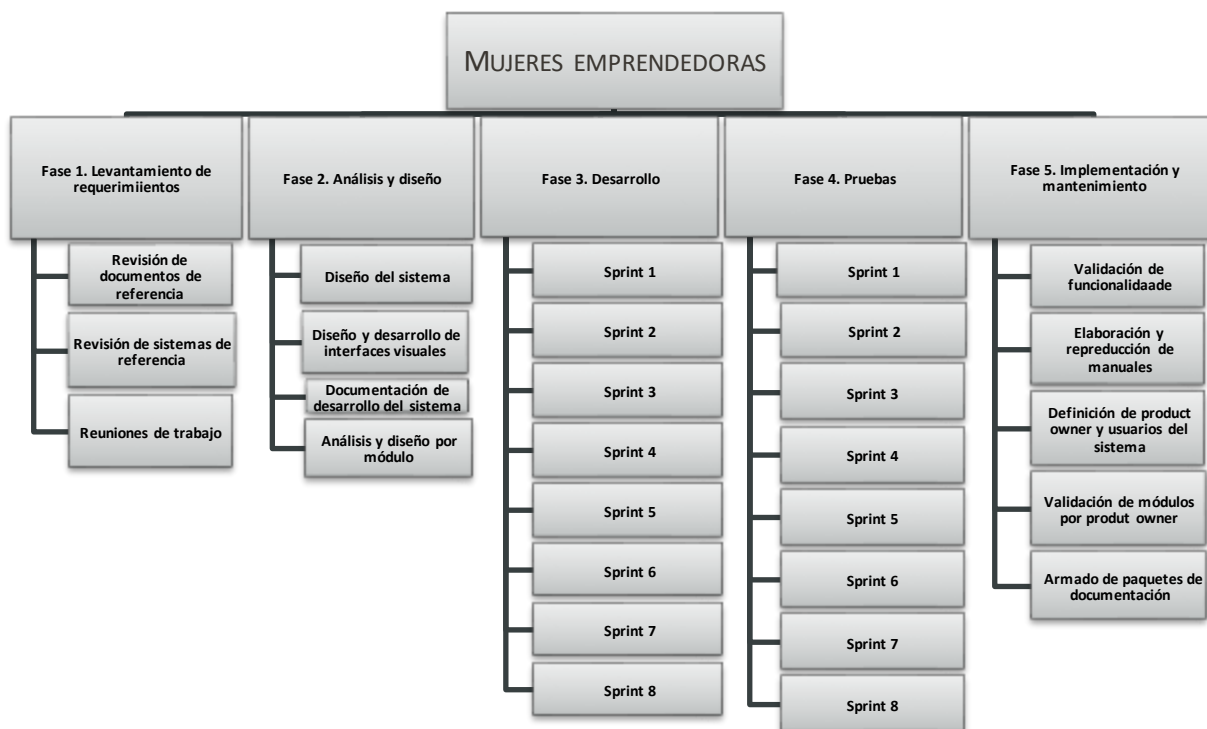
La tercera fase corresponde al desarrollo del software, se considera parte fundamental de la integración del sistema, debido a que se genera el código y se define la estructura de la base de datos, se implementan las funciones del sistema, se desarrollan las interfaces e incluso se habilita el servidor de pruebas.

La siguiente fase corresponde a las pruebas del software y su objetivo es asegurar que el sistema cumpla con las especificaciones requeridas por el cliente y se eliminen posibles errores.

La última fase es la de implementación, en la cual se validan las funcionalidades y se elaboran los manuales digitales. El Product Owner valida los módulos y se llevan a cabo las sesiones de asimilación para su entrega.

A continuación se presenta un mapa de nivel con los paquetes de trabajo, en el que se ejemplifican de forma concreta, ágil y esquemática las actividades que se ejecutan para integrar el sistema.

### Mapa a nivel de paquetes de trabajo



**Mapa 1.** Mapa a niveles de paquetes. Fuente: Elaboración propia

# PROCESOS DE GESTIÓN

## SUPOSICIONES, DEPENDENCIAS Y RESTRICCIONES

---

Las suposiciones mencionan los aspectos que no están claramente definidos para el sistema pero que de igual forma son asumidos para el desarrollo del proyecto, estas son:

- Los equipos de los usuarios del sistema cumplen con los requerimientos mínimos de hardware y software
- Los servidores en los que se monta el sistema cumplen con los requerimientos mínimos de hardware y software

### Restricciones

- Se accede al sistema a través de un nombre de dominio (DNS), vía un navegador web.
- Lenguajes y tecnologías en uso: HTML, Javascript, Java Enterprise Edition, CSS, SQL a través de MySQL
- El sistema debe ser capaz de atender consultas concurrentemente
- Los componentes utilizados en la programación del sistema son de código libre para no tener restricciones de licenciamiento y contar con soporte de la comunidad ante cualquier duda o problema de desarrollo
- Utiliza servidor web (Apache) para la recepción de solicitudes desde los clientes, así como proxy hacia los servicios implementados en los servidores de aplicaciones. Y como base de base de datos (MySQL)
- Utiliza los protocolo de comunicación HTTPS para la comunicación y transferencia de información entre módulos y sistemas
- La información que contiene el sistema viaja encriptada a través del uso de protocolo HTTPS, para evitar la lectura por parte de terceros durante su transferencia
- Para la información que se intercambia a través de los servicios web se recomienda utilizar el esquema JSON para su fácil lectura utilizando frameworks de desarrollo

# PROCESO TÉCNICO

## METODOLOGÍAS, TÉCNICAS Y HERRAMIENTAS

---

La convocatoria retos Jalisco, del programa de Mujeres jefas de familia requiere el desarrollo de un sistema que unifique los datos y permita la interoperabilidad de los distintos sistemas implicados en la administración y operación de proyectos, así como, tener una estructura de trabajo definida, basada en estándares que faciliten la información de diversas áreas en la institución, el escalamiento modular, la seguridad en la transferencia y uso de la información.

Para el desarrollo se propone el uso de tecnologías Open Source, que cuenten con comunidades de soporte y desarrollo establecidas, así como, la implementación de patrones de diseño Modelo-Vista-Controlador (mvc), y sistemas de persistencia de datos con casos de éxito en sistemas gubernamentales y empresariales.

### Componentes

La siguiente información muestra la descomposición del software de manera lógica en unidades más pequeñas con el fin de delimitar las responsabilidades y funcionalidades en componentes diferentes, con la intención de reducir las adversidades del sistema y favorecer el mantenimiento del mismo.

Los siguientes componentes son utilizados en el desarrollo, estas herramientas tecnológicas se han elegido con base a revisiones de sus características y por tratarse de software de código abierto.

- Apache
- HTML5
- CSS3
- Javascript
- Ajax
- PHP
- MySQL

A través de interfaces y APIs es posible el acceso e interacción con los documentos y sus diferentes versiones desde aplicaciones y desarrollos de terceros.

Adicionalmente, los elementos de *frameworks* y librerías que se están utilizando, son estándares en el ámbito de desarrollo y utilizados en gran cantidad de proyectos a nivel mundial, con soporte tanto por sus comunidades como por empresas que proporcionan consultorías en su uso e implementación.

## Normatividad aplicable

La totalidad de componentes del nuevo modelo deberá considerar las mejores prácticas de la industria, relacionadas con procesos, seguridad y disponibilidad, monitoreo y tecnologías de información, entre otros.

Para la documentación se utilizan diferentes metodologías (IEEE y SCRUM), las cuales están estipuladas en cada documento y de manera general se encuentran descritas.

El Instituto de Ingeniería Eléctrica y Electrónica (IEEE) es especialista en estandarización para el desarrollo de software, en este sentido, proporciona una estructura con los elementos mínimos que deben ser utilizados en la producción del sistema y contribuye a generar documentación de calidad.

Con la metodología SCRUM, se genera un marco de trabajo que define los roles y herramientas para facilitar la colaboración en equipo, alcanzar los objetivos planteados en periodos cortos y obtener los mejores resultados para el desarrollo del sistema.

La importancia de la alineación con estos estándares, radica en definir un marco normativo que permita, entre otras cosas, que las instituciones gubernamentales tengan claro la importancia estratégica de institucionalizar las prácticas de planeación, organización y operación bajo un mismo modelo, que facilite unificar arquitecturas y con ello la interoperabilidad y los servicios digitales, a través de la planeación estratégica de TICS.



# PLAN DE MANTENIMIENTO

El estándar IEEE 1219 [IEEE, 1993] define el Mantenimiento del Software como “la modificación de un producto software después de haber sido entregado [a los usuarios o clientes] con el fin de corregir defectos, mejorar el rendimiento u otros atributos, o adaptarlo a un cambio en el entorno”

En el estándar ISO 12207, de Procesos del Ciclo de Vida del Software [ISO/IEC, 1995] se establece que “el Proceso de Mantenimiento contiene las actividades y tareas realizadas por el mantenedor. Este proceso se activa cuando el producto software sufre modificaciones en el código y la documentación asociada, debido a un problema o a la necesidad de mejora o adaptación. El objetivo es modificar el producto software existente preservando su integridad. Este proceso incluye la migración y retirada del producto software. El proceso termina con la retirada del producto software”. El mantenedor es la organización que proporciona el servicio de mantenimiento

1. Introducción
2. Concepto de mantenimiento
3. Actividades de mantenimiento y de la organización
4. Recursos
5. Proceso (como se va a llevar a cabo el trabajo)
6. Entrenamiento
7. Registros e informes de mantenimiento

El Proceso de Mantenimiento contiene las actividades y tareas necesarias para modificar un producto software existente conservando su integridad. El Proceso de Mantenimiento debe activarse cuando existe un requerimiento de mantenimiento de un producto software. Tan pronto como se active el proceso, se deben desarrollar los Planes y Procedimientos de Mantenimiento y se deben asignar los recursos para el mantenimiento.

Después de que el producto software sea entregado (en caso de adjudicación), los responsables del área de mantenimiento modificarán el código y documentación asociados como respuesta a una petición de modificación o informe de problema.

El objetivo global del mantenimiento de software es modificar el producto software conservando su integridad. Este proceso da soporte al producto software desde su nacimiento pasando por la migración a otro entorno hasta su retiro. El proceso finaliza cuando el producto software es retirado.

Este proceso consistirá de las siguientes actividades:

- 1. Implementación del Proceso**
- 2. Análisis de Modificaciones y Problemas**
- 3. Implementación de Modificaciones**
- 4. Revisión/Aceptación del Mantenimiento**
- 5. Migración**

## PROCESO DE IMPLEMENTACIÓN

---

Esta actividad consiste de las siguientes tareas:

El mantenedor desarrollará, documentará y ejecutará los planes y procedimientos para realizar las actividades y tareas del Proceso de Mantenimiento.

Se establecerá el procedimiento para recibir, registrar y hacer seguimiento a los informes de problemas y requisitos de modificaciones hechas por los usuarios así como suministrar retroalimentación a los usuarios. Siempre que se encuentren problemas, estos se deben registrar e ingresar al Proceso de Solución de Problemas (ver numeral 6.8).

Se implementará el Proceso de Administración de la Configuración para manejar las modificaciones al sistema existente.

## ANÁLISIS DE PROBLEMAS Y MODIFICACIONES

---

Esta actividad consiste de las siguientes tareas:

Se analizará el impacto que el informe del problema o requisito de modificación puedan tener sobre la organización, sistema existente y el enlace de sistema buscando:

- a. Tipo; por ejemplo correctivo, mejora, preventivo o adaptación a un nuevo ambiente;
- b. Alcance; por ejemplo tamaño de la modificación, costo, tiempo;
- c. Criticidad; por ejemplo impacto en el desempeño, inviolabilidad o seguridad.

Se deberá replicar o verificar el problema. Basado en el análisis, el mantenedor debe desarrollar opciones para implementar la modificación.

Se deberá documentar el requisito de problema/modificación, los resultados del análisis y opciones de implementación.

Se obtendrá la aprobación de la opción de modificación seleccionada según se especifique en el proyecto o contrato.

## IMPLEMENTACIÓN DE MODIFICACIONES

---

Esta actividad consiste de las siguientes tareas:

Se realizará un análisis para determinar qué documentación, unidades de software y versiones se deben modificar. Estos se deben documentar.

Al ingresar al Proceso de Desarrollo para implementar las modificaciones, se deberán seguir los requisitos como sigue:

- 1) Se debe definir y documentar los ensayos y criterios de evaluación para ensayar y evaluar las partes modificadas y las no modificadas (unidades de software, componentes e ítems de configuración) del sistema.

- 2) La correcta y completa implementación de los requisitos nuevos y modificados se debe asegurar. También se debe asegurar que los requisitos originales no modificados no se hayan afectado. Los resultados de los ensayos se deben modificar.

## REVISIÓN/ACEPTACIÓN DEL MANTENIMIENTO

---

Esta actividad consiste de las siguientes tareas:

Se deberán conducir revisiones con el líder de proyecto que autoriza la modificación para determinar la integridad del sistema modificado.

Se obtendrá la aprobación para completar la modificación satisfactoriamente según se especifique en el proyecto o contrato.

## MIGRACIÓN

---

En caso de migración del sistema o producto de software (induyendo datos) deberá migrar de un ambiente operacional viejo a uno nuevo, se debe asegurar que cualquier producto de software o dato producido o modificado durante la migración se encuentren en concordancia con esta directiva.

Los ítems incluidos en el plan de migración deben comprender:

- a. Análisis de requisitos y definición de migración;
- b. Desarrollo de herramientas para la migración;
- c. Conversión de productos de software y datos;
- d. Ejecución de la migración;
- e. Verificación de la migración;
- f. Soporte para el antiguo ambiente en el futuro.

Los usuarios serán notificados sobre los planes y actividades de migración. En caso de ser necesaria.

## PROCESO DE MANTENIMIENTO

---

El proceso del ciclo de vida Mantenimiento de Software empieza con la implementación de éste proceso donde se planifica el mantenimiento y acaba con la retirada del producto. Incluye la modificación de código y documentación debido a algún problema o la necesidad de mantenimiento.

El objetivo del proceso de mantenimiento será modificar el sistema preservando siempre su integridad. Se tendrá en cuenta los requerimientos implícitos y las restricciones impuestas sobre el desarrollador original.

El proceso de mantenimiento podrá ser:

- Adaptativo: Modificación de un producto software, después de su entrega, para conseguir que sea utilizable en un nuevo entorno.
- Correctivo: Modificación reactiva de un producto software, después de su entrega, para corregir defectos detectados. Errores actuales detectados.
- Perfectivo: Modificación de un producto software, después de su entrega, para mejorar su rendimiento o su mantenibilidad.
- Preventivo: Modificación de un producto software, después de su entrega, para detectar y corregir defectos latentes antes de que produzcan fallos efectivos.

## NIVEL DE SOPORTE TECNOLÓGICO

En caso de que el sistema sufra algún inconveniente que requiera cambios en el mismo, es necesario seguir el protocolo de mantenimiento, el cual indica que el primer paso es notificar al departamento de desarrollo para que realicen los cambios de software, se ingrese al servidor, se detenga, se suban los cambios, y se vuelva a iniciar para que los usuarios lo continúen utilizando.

Para notificar de manera correcta al departamento de desarrollo el cliente deberá llenar el formato “Bitácora de incidentes” que se adjunta a este documento.

### INCIDENTE

---

Al momento de presentarse el incidente, el usuario deberá tener a la mano y llenar de forma detallada la “bitácora de incidentes” esto con el fin de contar con la información más detallada y poder dar una mejor solución al mismo.

### PROCESO DE NOTIFICACIÓN

---

Una vez debidamente llenada la bitácora de incidente, está deberá ser enviada al correo de soporte técnico [grupo.sitti@gmail.com](mailto:grupo.sitti@gmail.com). El incidente se atenderá conforme a los SLA’s de atención que se acordarán una vez adjudicado el servicio.

### MANTENIMIENTO O CORRECCIONES

---

Una vez que el departamento de desarrollo identifique la causa del inconveniente reportado y se le otorgué al cliente una hora estimada para su atención, se dará mantenimiento al sistema, subiendo una nueva versión del mismo al servidor, lo cual podría causar mínimos tiempos fuera de servicio del sistema, conforme al Plan de Mantenimiento, mencionado en este Plan de Trabajo.