



Centro de Investigación en Matemáticas, A.C.

---

---

CIMAT

# Mantenimiento Ágil en Cloud Computing: Disponibilidad del Servicio

## REPORTE TÉCNICO

Que para obtener el grado de

**Maestro en Ingeniería de Software**

P r e s e n t a

**Oscar Carrillo Razón**

Director de Reporte Técnico  
**Dr. Hugo A. Mitre Hernández**

Zacatecas, Zacatecas., 30 de 07 de 2013



## Índice

Resumen .....	1
<b>1 INTRODUCCIÓN .....</b>	<b>1</b>
<b>2 TRABAJOS RELACIONADOS .....</b>	<b>3</b>
2.1 Trabajos Relacionados a la Mejora de Procesos y Mantenimiento.....	3
2.2 Trabajos Relacionados a los Problemas en la Disponibilidad de los Servicios.....	4
2.2.1 Tolerancia a Fallos .....	4
2.2.2 Balanceo de Cargas.....	5
2.2.3 Replicación de Datos .....	5
<b>3 DESCRIPCIÓN DE LA METODOLOGÍA DE TRABAJO.....</b>	<b>5</b>
<b>3.1 FASES DE TRABAJO.....</b>	<b>5</b>
<b>3.2 ORGANIZACIÓN ESTUDIADA.....</b>	<b>5</b>
<b>4 SELECCIÓN DE MODELOS PARA LA OBTENCIÓN DE PRÁCTICAS.....</b>	<b>6</b>
4.1 S3M – Modelo de Madurez del Mantenimiento Del Software .....	6
4.2 ITIL – V3 Diseño del Servicio .....	7
4.3 OWASP (Proyecto Abierto de Seguridad de Aplicaciones Web) - Guía de Desarrollo .....	7
<b>5 IDENTIFICACIÓN Y REDUCCIÓN DE PRÁCTICAS PARA DISPONIBILIDAD Y MÉTODOS ÁGILES.....</b>	<b>7</b>
5.1 Primer Filtro (Mantenimiento).....	8
5.2 Segundo Filtro (Disponibilidad) .....	8
5.3 Tercer Filtro (Manifiesto Ágil).....	9
<b>6 INTEGRACIÓN Y MEJORA DE LAS PRÁCTICAS PARA EL PROCESO DE MANTENIMIENTO.....</b>	<b>11</b>
6.1 Mejores Prácticas + Enfoque .....	13
6.2 Validación de las Prácticas Resultantes.....	13
6.2.1 Etapa 1 .....	13
6.2.2 Etapa 2.....	13
6.3 Resumen de Prácticas .....	13
6.4 Clasificación de Prácticas .....	14
6.5 Adaptación de Prácticas.....	15
6.6 Selección de Herramientas de Soporte Para las Prácticas.....	15
<b>7 PROPUESTA DE IMPLEMENTACIÓN DEL PROCESO ÁGIL DE GESTIÓN DE MANTENIMIENTO PARA ENTORNOS CLOUD ENFOCADO EN DISPONIBILIDAD.....</b>	<b>16</b>
7.1 Análisis de la Organización .....	16
7.1.1 Proceso Actual de Desarrollo .....	16
7.1.2 Proceso Actual de Mantenimiento.....	17
7.2 Propuesta de Proceso Ágil De Gestión del Mantenimiento .....	18
7.3 Cobertura de la propuesta en el proceso actual .....	19
<b>8 CONCLUSIONES Y TRABAJO FUTURO.....</b>	<b>20</b>
<b>REFERENCIAS .....</b>	<b>21</b>
<b>ANEXOS .....</b>	<b>23</b>

## Índice de Figuras

FIG. 1 APLICACIÓN DE FILTROS EN LA REDUCCIÓN DE PRÁCTICAS.....	8
FIG. 2 PROCESO DE INTEGRACIÓN Y MEJORA DE LAS PRÁCTICAS PARA EL PROCESO DE MANTENIMIENTO. ....	11
FIG. 3 ÚLTIMA VERSIÓN DEL PROCESO DE DESARROLLO DE SOFTLOGIK. ....	17
FIG. 4 PROCESO ACTUAL DE MANTENIMIENTO PARA ATENCIÓN Y SOPORTE AL CLIENTE DE SOFTLOGIK. ....	18
FIG. 5 .PROPUESTA DE INTEGRACIÓN DE PRÁCTICAS A MY AGILE BOARD PARA EL SEGUIMIENTO Y DESARROLLO DE LA ACTIVIDAD. ....	18
FIG. 6 .GRÁFICA DE COBERTURA DE LAS PRÁCTICAS. ....	20
FIG. 7 SECUENCIA, RELACIÓN E INTERACCIÓN ENTRE LAS PRÁCTICAS, ENFOQUES Y HERRAMIENTAS FINALES. (PARTE 1/2).....	31
FIG. 8 SECUENCIA, RELACIÓN E INTERACCIÓN ENTRE LAS PRÁCTICAS, ENFOQUES Y HERRAMIENTAS FINALES. (PARTE 2/2).....	32
FIG. 9 SUBPROCESOS DE LA ACTIVIDAD REALIZAR ANÁLISIS DE RENDIMIENTO.....	33
FIG. 10 SUBPROCESOS DE LA ACTIVIDAD REALIZAR ANÁLISIS DE CONFIGURACIÓN.....	33

## Índice de Tablas

TABLA 1 REDUCCIÓN DE PRÁCTICAS ITIL Y OWASP (FILTRO 1: MANTENIMIENTO).....	9
TABLA 2 REDUCCIÓN DE PRÁCTICAS S3M, ITIL Y OWASP (FILTRO 2: DISPONIBILIDAD).....	10
TABLA 3 REDUCCIÓN DE PRÁCTICAS S3M, ITIL Y OWASP (FILTRO 3: MANIFIESTO ÁGIL) .....	12
TABLA 4 ENFOQUES DE LAS PRÁCTICAS.....	13
TABLA 5 TÉRMINOS CORTOS DE LAS PRÁCTICAS .....	14
TABLA 6 CLASIFICACIÓN DE LAS PRÁCTICAS.....	14
TABLA 7 ADAPTACIÓN ENTRE LAS PRÁCTICAS.....	15
TABLA 8 RUTA DE SEGUIMIENTO DE LAS PRÁCTICAS.....	19
TABLA 9 PROPUESTA DE COBERTURA DE PRÁCTICAS NO CUBIERTAS O PARCIALES .....	20
TABLA 10 DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A S3M.....	23
TABLA 11 DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A ITIL .....	24
TABLA 12 DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A OWASP .....	24
TABLA 13 RESULTADOS DE LA VALIDACIÓN DE LAS PRÁCTICAS EN LA ORGANIZACIÓN.....	26
TABLA 14 LISTADO MEJORES PRÁCTICAS FINALES.....	30

## Índice de Anexos

ANEXO 1 RESULTADOS DE APLICACIÓN DE LOS FILTROS .....	23
ANEXO 2 RESULTADOS DE LA VALIDACIÓN DE LAS PRÁCTICAS EN LA ORGANIZACIÓN .....	26
ANEXO 3 BENCHMARKING DE HERRAMIENTAS DE SOPORTE.....	29
ANEXO 4 RESULTADOS OBTENIDOS DE LA APLICACIÓN DEL PROCESO DE INTEGRACIÓN Y MEJORA DE LAS PRÁCTICAS FINALES .....	30

# Mantenimiento Ágil en Cloud Computing: Disponibilidad del Servicio

**Resumen**— Dentro del cómputo en la nube o “Cloud Computing” se utilizan recursos de cómputo donde se proveen servicios a terceros a través del internet y se hace por medio de diferentes modelos de servicio y capas de abstracción (Servicio, Plataforma e Infraestructura). La implementación de un sistema de gestión del mantenimiento entre las capas que integran el “Cloud Computing” es muy importante para lograr cumplir con los requerimientos de disponibilidad de la información y de los servicios que se ofrecen a través de este medio. Cuando se ofrecen servicios a través del “Cloud Computing” se tienen problemas que afectan directamente la disponibilidad de los servicios, con respecto a esto, se identificaron tres problemas principales: (1) falta de soporte a tolerancia a fallos, (2) Limitación de recursos para el balanceo de cargas y (3) falta de control en la replicación de datos; estos problemas requieren ser considerados dentro de un adecuado sistema de gestión del mantenimiento, por lo que se propone la definición de un proceso de gestión de mantenimiento basado en mejores prácticas de **S3M** (Software Maintenance Maturity Model), **ITIL** (Information Technology Infrastructure Library) y **OWASP** (The Open Web Application Security Project) alineadas al manifiesto ágil y enfocadas en la disponibilidad para implementar el proceso de manera ágil y garantizar la disponibilidad de los servicios que se ofrecen en una empresa de TI (Tecnologías de la Información).

**Palabras Clave**— Mantenimiento, Cloud Computing, Disponibilidad, tolerancia a fallas, balanceo de cargas, Replicación de datos, S3M, ITIL, OWASP.



## 1 INTRODUCCIÓN

El cómputo en la nube mejor conocido como “Cloud Computing” (CC) utiliza recursos de cómputo donde los servicios se proveen a través de la internet usando diferentes modelos de servicio y capas de abstracción[1]. Dentro de los principales modelos de servicio que integran el CC se encuentran: “Infraestructura as a Service”(IaaS), “Plataforma as a Service”(PaaS) y el más común, el “Software as a Service”(SaaS)[2], [3], los cuales están distribuidos en las capas de: Infraestructura (Unidades físicas), Plataforma (Ambientes de operación) y Software (Aplicaciones)[4].

Una de las principales características de los entornos CC con respecto a los servicios ofrecidos, es que si se ofrecen servicios por este medio, se debe garantizar su disponibilidad para cumplir con los requerimientos de los clientes, los acuerdos de nivel de servicio (SLA por sus siglas en inglés) ofrecidos y la continuidad de los servicios[4].

La disponibilidad, es definida como la capacidad que tiene un sistema para limitar, controlar y proveer un servicio adecuado bajo ciertas restricciones que se presenten[5]. En CC, la disponibilidad puede referirse al tiempo de vida de un sistema, que puede ser dividido en un conjunto de “up states” (estados activo) y un conjunto de “down states” (estados caído), en el que la disponibilidad puede ser categorizada como *disponibilidad instantánea* (Probabilidad que tiene un sistema de que esté activo al iniciarse), *disponibilidad estable* (Probabilidad que tiene un sistema de estar activo por un periodo de tiempo) y

disponibilidad inherente (Porcentaje de tiempo esperado durante el cual se realizan las funciones requeridas) [6].

La disponibilidad de los servicios en entornos CC se puede ver afectada por diferentes problemas, entre los problemas más relevantes que la afectan se encuentran:

- La *tolerancia a fallas*, que puede ser una de las mayores preocupaciones para garantizar la disponibilidad de servicios críticos[7].
- *Limitación de recursos*, aplicación de *balanceo de cargas* para asegurar la disponibilidad de la red[8].
- *Replicación de datos*, saber qué datos replicar, cuando replicarlos, cuantas replicas hacer y donde colocarlas para reducir tiempos de espera e incrementar velocidad de acceso[6].

Estos problemas están enfocados a aplicarse tanto en recursos de software como hardware, en el CC se ofrecen servicios y recursos basados en estos 2 tipos de recursos (hardware y software) que permiten a los usuarios hacer uso de ellos tal cual están y bajo demanda, es decir, los servicios ofrecidos son en base a productos de hardware configurado y software funcional terminado de acuerdo a las necesidades de los clientes [4], [9], por lo que la etapa del ciclo de vida de desarrollo de software que mejor se adecúa para buscar satisfacer estos problemas es la etapa de mantenimiento, ya que el mantenimiento se define como “El hecho de mantener una entidad en un estado actual de reparación eficiente, o validado, para preservarla de un fallo o deterioro”[10], en el área de software se define como “los cambios relacionados con el sistema de software después de la entrega”[10], [11] y pueden ser de 4 diferentes tipos de mantenimiento[12]:

- 1) Correctivo: Cambios para corregir errores,
- 2) Adaptativo: Cambios para adaptar a un ambiente diferente,
- 3) Perfectivo: Cambios basados en mejora y nuevos requerimientos de los usuarios y
- 4) Preventivo: Cambios para prevenir problemas.

Para las capas del CC (Plataforma, Infraestructura y Software) el mantenimiento es una etapa primordial en el aseguramiento de la continuidad del servicio ya que de esto depende mantener los servicios disponibles, libres de errores y optimizados de acuerdo a las necesidades de los clientes [13],[14]. De forma general el mantenimiento comúnmente excede del 50% del costo total del ciclo de vida de los proyectos [15] y específicamente en el mantenimiento de software un estimado del 60 al 80 por ciento de los recursos son usados para este fin[10]. El mantenimiento comúnmente en las pymes se lleva a cabo sin ningún control o proceso definido y están basados en el conocimiento empírico obtenido a través de llevar a cabo sus actividades diarias dentro de la organización [11].

Un adecuado sistema de gestión del mantenimiento debe satisfacer todos los criterios de los servicios al cliente, de los usuarios y de los del dominio técnico; donde los criterios son los procesos, técnicas y herramientas usadas para satisfacer a los clientes y usuarios y la disponibilidad del personal de mantenimiento para ayudar en la mejora, implantación y definición del proceso [16].

Los procesos se han posicionado como elementos primordiales en la mejora de las organizaciones ya que son usados para llevar a cabo sus actividades diarias y es por medio de los procesos como se puede gestionar el trabajo de las organizaciones de manera eficaz y eficiente [17].

Dentro de la mejora de procesos, se encuentra problemas durante su implementación, como es la resistencia al cambio por parte del personal de la organización[17]. Para afrontar estas situaciones, han surgido metodologías para la implementación de mejoras de procesos de forma gradual y enfocada en la disminución de la resistencia al cambio como es la metodología MIGME-RRC[18] de la cual, uno de sus puntos propone la identificación de las mejores prácticas internas de la organización para ayudar a comprender su forma de trabajo.

Hoy en día, las prácticas ágiles son comúnmente utilizadas en las pymes, por la simplicidad que les ofrecen durante sus procesos para la gestión y control de los proyectos, nuevos enfoques aplicados usando los principios ágiles a las diferentes etapas de los proyectos han surgido, tal es el caso de 2 organizaciones de TI canadienses (Una enfocada en aplicaciones para e-commerce y sistemas de radares terrestres y la otra dedicada a consultoría de gestión de TI ) que utiliza prácticas ágiles en el mantenimiento utilizando Scrum y XP para la mejora de sus procesos [19].

El propósito de este estudio consistió en definir un

proceso de gestión de mantenimiento que pueda ser tomado como una propuesta inicial de mejora y ser implementado de manera ágil de acuerdo al proceso actual de mantenimiento y desarrollo de software de Softlogik (una empresa de consultoría en Tecnologías de la Información (TI)) en el área de “Cloud Computing” para garantizar la disponibilidad de los productos y de los servicios que se ofrecen a través de este medio debido a que se estima un incremento considerable en los servicios tanto de mantenimiento como de soporte al lanzar al mercado sus nuevos productos de SaaS los cuales aún tienen en fase de pruebas.

Si se logra definir un adecuado proceso de gestión del mantenimiento aplicado a las capas que conforman el CC enfocándolo en disponibilidad, se logrará incrementar la disponibilidad de estos entornos y de esta forma satisfacer los criterios tanto técnicos como de los servicios que se ofrecen a los clientes.

En esta investigación se considera la importancia e impacto del mantenimiento entre las diferentes capas que integran el CC; donde el mantenimiento es importante para lograr cumplir con los requerimientos de niveles aceptables de disponibilidad de la información y servicios que se pueden ofrecer a través del CC. Para lo que se realiza un análisis de las principales prácticas que ofrece el modelo de madurez de mantenimiento de software **S3M** (del inglés Software Maintenance Maturity Model) [16] en sus primeras 3 fases, la 0,1 y 2 para establecer un proceso de gestión de mantenimiento y la integración de prácticas específicas de la Biblioteca de Infraestructura de Tecnologías de Información **ITIL**(del inglés Information Technology Infrastructure Library)[13] y del Proyecto abierto de seguridad de aplicaciones web **OWASP** (del inglés Open Web Application Security Project)[20] para reforzar la satisfacción del atributo de calidad de disponibilidad en los servicios del CC.

El análisis realizado para generar la propuesta consiste en 3 filtros, el primer filtro, consiste en extraer las mejores prácticas de los modelos alineándolos a prácticas de mantenimiento, el segundo filtro se enfoca en alinearlos a prácticas que permitan asegurar la disponibilidad en el CC, y como tercer y último filtro, se extraen solo las mejores prácticas resultantes del segundo filtro y se alinean al manifiesto ágil. En base al resultado de este análisis se genera como solución inicial una propuesta de proceso de gestión de mantenimiento ágil ajustado al proceso actual de mantenimiento y desarrollo de software de Softlogik para enfocarlo en garantizar la disponibilidad de los productos y de los servicios que son ofrecidos por la organización; Donde los procesos actuales fueron obtenidos por medio de la identificación de las prácticas internas de softlogik para comprender su forma de trabajo como lo sugiere la metodología MIGME-RRC.

Este trabajo está estructurado de la siguiente manera: en la sección 2 se analizan trabajos relacionados a la disponibilidad de servicios en entornos CC para identificar

problemáticas referentes; en la sección 3 se presentan brevemente los 3 modelos utilizados para elaborar este análisis ; en la sección 4 se presenta el proceso realizado en la aplicación de los filtros, en la sección 5 se presenta el proceso realizado para la integración y la mejora de las prácticas resultantes de los filtros, y en la sección 6 se genera una propuesta para la implementación de las prácticas resultantes del análisis en el proceso actual de desarrollo y de mantenimiento de softlogik tomando como base las áreas de gestión del proceso de mantenimiento de la norma s3M.

## 2 TRABAJOS RELACIONADOS

### 2.1 TRABAJOS RELACIONADOS A LA MEJORA DE PROCESOS Y MANTENIMIENTO

En este punto se presentan algunos trabajos relacionados a la mejora de procesos en el área de mantenimiento principalmente en el uso de metodologías ágiles. El estudio realizado a estos trabajos permitió identificar que las principales técnicas o metodologías de trabajo utilizadas cuando se trata de aplicar agilidad son: las prácticas de Scrum, las prácticas esbeltas o Lean con el uso de Kanban, y programación extrema XP (Extreme Programming).

Los procesos se han posicionado como elementos primordiales en la mejora de las organizaciones ya que a través de ellos es como se pueden diagnosticar los problemas que permiten identificar mejoras en los procesos y las estructuras de las organizaciones y son usados para llevar a cabo sus actividades diarias además de que es por medio de los procesos como se puede gestionar el trabajo de las organizaciones de una manera eficaz y eficiente [17].

Dentro de la mejora de procesos, se encuentra problemas durante su implementación, como es la resistencia al cambio por parte del personal de la organización[17]. Para afrontar estas situaciones, han surgido metodologías para la implementación de mejoras de procesos de forma gradual y enfocada en la disminución de la resistencia al cambio como es la metodología MIGME-RRC que permite la implementación gradual de mejoras de los procesos de una organización, ya que su proceso permite tomar la capacidad de cambio de la organización para disminuir la resistencia al cambio. La metodología está dividida en 4 etapas[21]:

- 1-Identificar mejores prácticas internas.
- 2-Evaluar el rendimiento de la organización.
- 3-Analizar las prácticas externas.
- 4.-Implementar mejoras en los procesos.

En la parte de mejora de procesos, se identificaron trabajos relacionados a la mejora de los procesos de mantenimiento usando prácticas ágiles.

Dentro de los trabajos relacionados, se encuentra el caso de Ericsson R&D una empresa de TI [22] que implementa Kanban en el mantenimiento de un producto de

telecomunicaciones, el cual es implementado en 2 áreas de mantenimiento de la organización ubicadas en diferentes sitios, la de Soporte a las peticiones del cliente CSR (Por sus siglas en inglés Customer Support Request) y el área de manejo de errores.

En esta implementación también se consideraron mejores prácticas de Scrum para complementar la implementación del Kanban. Se implementaron los principios de Lean con el Kanban para visualizar el flujo de trabajo, establecer el trabajo en progreso y medir y optimizar el tiempo de ejecución. Por parte de Scrum, se adoptaron las reuniones diarias y las retrospectivas. Los principales beneficios obtenidos fueron en el trabajo en equipo, su responsabilidad, se cambió el trabajo a tirar (“pull”) en lugar de empujar (“push”), los equipos desafiaron la zona de confort, mayor cooperación entre las áreas de trabajo, el equipo adquirió una mentalidad de mejorar continuamente y se obtuvo una mejor visibilidad de cómo se lleva el proyecto.

Otro caso que presenta el uso de Kanban es el de la British Broadcasting Corporation (BBC) en Londres [23] que muestra cómo se mejora el rendimiento de los equipos mediante la aplicación de los principios lean en la gestión de proyectos de software, el estudio se aplica a un equipo (9 personas) de la BBC que participan en un proyecto de desarrollo de software. El proyecto estaba enfocado en desarrollar un nuevo software y el mantenimiento de software en el que la mayoría de las solicitudes de fallo recibidas eran causadas por fallas en el mismo sistema.

El enfoque fue aplicado a mejorar las prácticas de: (1) pruebas unitarias, (2) pruebas de aceptación automática, (3) control de códigos fuente, (4) Desacoplamiento para mejorar el código legado y (5) el concepto MMF (Mínimo de funciones negociables por sus siglas en inglés Minimum Marketable Feature Concept).

En la aplicación del Kanban para la gestión del proyecto, se redujo el WIP (trabajo en progreso por sus siglas en inglés, “Work In Progress”) para controlar el trabajo, lo que proporcionó mayor visibilidad del proyecto, flexibilidad y productividad.

La aplicación del enfoque lean mediante el Kanban en el proceso de software impacto en la reducción de: tiempo de ejecución, tiempo de desarrollo, cantidad errores, retrasos de las entregas y un impacto en el incremento en la frecuencia de las entregas, lo que permitió identificar que las entregas pequeñas de manera frecuente reducen los riesgos en el proyecto y son de alto valor.

Otro caso de estudio en el que aplican metodologías ágiles en 2 organizaciones de TI canadienses (Una enfocada en aplicaciones para e-commerce y sistemas de radares terrestres y la otra dedicada a consultoría de gestión de TI ) que utilizan prácticas ágiles en el mantenimiento utilizando Scrum y XP para la mejora de sus procesos[19].

A diferencia de los casos anteriores, Anderson et al. [24] presenta un caso ocurrido en el equipo XIT (uno de los 8 grupos de TI de Microsoft) en el que hace uso de los 5 pasos de la Teoría de Restricciones TOC (por sus siglas en inglés Theory of Constraints) para mejorar el equipo debido a que era considerado que tenía el peor rendimiento en comparación con los otros equipos de Microsoft y sus clientes no estaban contentos. Este equipo daba soporte a más de 80 aplicaciones internas en todo el mundo atendiendo solicitudes de cambio y corrección de errores. La aplicación de esta teoría, permitió el incremento de la productividad en más del 100%.

Se hicieron 3 aportaciones a la mejora del equipo XIT haciendo uso de TOC, estas aportaciones son: (1) aplicar Buffers, (2) No hacer estimaciones y (3) Reubicación de los recursos.

- (1) *La aplicación de los buffers* permitió la aplicación de las otras 2 aportaciones restantes. Antes de integrar los buffers, el equipo no tenía control sobre su capacidad de entrega y las nuevas solicitudes consumían mucho tiempo ya que tenían que entregar estimaciones dentro de las próximas 24hrs de acuerdo a su SLA (Acuerdo de nivel de servicio, por sus siglas en inglés Service Level Agreement). La introducción de los buffers permitió que se hiciera la estimación y se comprometiera con una fecha de entrega solo para los servicios que entraban al buffer y no a toda solicitud que llegaba al backlog.
- (2) *No hacer estimación*, la estimación tomaba aproximadamente el 40% de la capacidad del equipo, por lo que eliminar la estimación antes del buffer produjo un incremento en la productividad de forma inmediata, sin embargo, realizar este cambio implicó cambios de mentalidad de los clientes, en la dirección interna, y por consecuencia en su SLA.
- (3) *Reubicación de recursos*, aplicación de una razón de programadores y testers de 2:1, por lo que en el equipo finalmente quedo de 4:2 y no como inicialmente se estaba considerando de 3:3, esto para equilibrar el trabajo ya que se identificó que los testers tenían periodos bajos de trabajo.

Lu Gan et al. [25] describe un sistema de mantenimiento basado en CC, en el cual divide la estructura en 3 elementos principales, 1) Usuario, 2) Servidor y 3) Equipamiento, para el cual define una estructura de composición de 4 capas físicas,

- 1) Capa de Hardware, que incluye atender el hardware y el software ofrecido.
- 2) Capa de Plataforma, es el lugar donde todos los componentes operan y proveen las funciones a los usuarios, permite atender directamente los datos, la operación de las aplicaciones y gestión de los datos y el sistema.

- 3) Capa de Aplicación y 4) Capa de Usuario. Es donde se hace uso de los equipos y se transfiere la información y se atiende por medio de un panel de usuarios.

Este enfoque de mantenimiento puede ser usado en detección de errores, asistir en la toma de decisiones y el pronóstico de errores[25].

## 2.2 TRABAJOS RELACIONADOS A LOS PROBLEMAS EN LA DISPONIBILIDAD DE LOS SERVICIOS

En esta sección se presenta la revisión de algunos trabajos que fueron realizados anteriormente por otros autores, estos trabajos fueron realizados en su momento para satisfacer ciertas necesidades enfocadas en la disponibilidad de los servicios como la aplicación de un modelo de tolerancia a fallos, técnicas de balanceo de cargas y replicación de datos en entornos Cloud.

### 2.2.1 TOLERANCIA A FALLOS

Anju Bala et al. [7] en su artículo menciona que para minimizar el impacto de las fallas en los sistemas y su ejecución, las fallas deben ser anticipadas y manejadas de forma proactiva; para lo que se puede hacer uso de técnicas de tolerancia a fallas para predecirlas y tomar acciones antes de que ocurran; también se discuten algunas técnicas de tolerancia a fallas, estas técnicas se dividen en dos categorías, la primera se refiere a técnicas de tolerancia a fallas de forma reactiva y la segunda a técnicas de tolerancia a fallas de forma proactiva. A continuación se muestran algunas técnicas de cada una de las categorías.

- **Tolerancia a Fallas reactiva** para reducir el efecto del fallo cuando ocurre.
  - a) Checkpointing/restart
  - b) Replication
  - c) Job Migration
  - d) SGuard
  - e) Retry
  - f) Task Resubmission
  - g) User defined exception handling
  - h) Rescue workflow
- **Tolerancia a Fallas proactiva**, cuyo objetivo es permitir la recuperación de errores y fallos.
  - a) Software rejuvenation
  - b) Self Healing
  - c) Preemptive migration

Anju Bala et al. en su propuesta hace uso de una arquitectura de sistemas Cloud virtuales usando la herramienta HAProxy para enfrentar los temas relacionados con las caídas de los servidores a causa de los fallos, con esto demuestra que se puede asistir a las aplicaciones en los servidores para recuperarse de las fallas en milisegundos y con una mínima carga de procesamiento.



### 2.2.2 BALANCEO DE CARGAS

El balanceo de cargas es un proceso de mejora de rendimiento de un sistema ya sea paralelo o distribuido por medio de la redistribución del trabajo entre los diferentes procesadores o nodos de procesamiento [26].

El método de balanceo de cargas es aplicado a través de diferentes "Data Centers" para asegurar la disponibilidad de la red, reduciendo el uso de hardware, fallos de software y mitigar las limitaciones de los recursos, presenta Zenon Chaczko et al. [8] en su artículo, en el que también discute sobre el balanceo de cargas en "Cloud Computing" aplicado a un sistema basado en una solución de gestión de bases de datos para un Hospital con el objetivo de incrementar la disponibilidad y eficiencia en la recuperación de la información de múltiples bases de datos en diferentes entornos Cloud.

Zenon Chaczko et al. Proponen un proceso específico para balanceo de cargas y el uso de 2 métodos para mejorar la eficiencia del entorno, el "Proactive Scaling" para tener disponibles los recursos cuando más se necesitan y el "Compression" para tener acceso a grandes volúmenes de datos e incrementar tasas de transferencia. Además incorporan a su proyecto la gestión de tráfico de direcciones IP para determinar las direcciones de los servidores más cercanos y con menos carga de procesamiento[8].

### 2.2.3 REPLICACIÓN DE DATOS

La replicación de datos permite reducir tiempos de espera, incrementar velocidades de acceso a los datos e incrementar la disponibilidad de la información por medio de la provisión de diferentes réplicas de un mismo servicio, y todas estas replicas bajo un estado coherente [27].

Da-Wei et al.[6] mencionan que la replicación de datos es el principal mecanismo utilizado en Cloud para reducir el tiempo de espera del usuario, incrementando la disponibilidad y minimizando el consumo de banda ancha. En su artículo presentan una propuesta en la que incluye un modelo matemático para describir la relación entre la disponibilidad del sistema y el número de réplicas que deben hacerse considerando el tamaño, tiempo de acceso y la probabilidad de fallas que tiene cada archivo de datos; con el modelo matemático también se hace la identificación de archivos de los datos que son más populares en uso para darles prioridad de replicación y las réplicas son ubicadas usando nodos de una forma balanceada. En base al análisis de este modelo matemático presentan un algoritmo como estrategia de replicación de datos dinámica llamada "Dynamic Data Replication Strategy" (D2RS). Esta estrategia consta de 3 fases importantes:

1. Que datos deben ser replicados y cuando.
2. Cuantas replicas deben ser creadas en la nube y
3. Donde se deben ubicar las réplicas generadas

Los resultados experimentales realizados demostraron la eficiencia y la eficacia del sistema de mejora en la nube

presentado por la estrategia propuesta comentan Da-Wei et al.

## 3 DESCRIPCIÓN DE LA METODOLOGÍA DE TRABAJO

En esta sección se describe el método de trabajo utilizado. Las fases de trabajo usadas para la investigación se describen en la sección 3.1 y en la sección 3.2 se describe la organización seleccionada para el estudio.

### 3.1 FASES DE TRABAJO

Para llevar a cabo esta investigación, se llevaron a cabo diferentes pasos para la definición del proceso de gestión de mantenimiento y la propuesta para su implementación. Estos pasos son descritos a continuación:

Como primer paso se realiza la *Selección de Modelos para la Obtención de las Prácticas* de acuerdo a las necesidades de definir un proceso de gestión de mantenimiento para CC en el cual se considere el uso de Infraestructura y aplicaciones de Software basados en Web.

Como segundo paso se realiza la *Identificación y Reducción de Prácticas para Disponibilidad y Métodos Ágiles* por medio de filtros aplicados a las prácticas de los modelos seleccionados para alinearlos a actividades de mantenimiento, a actividades que garanticen la disponibilidad del CC y a actividades que se puedan aplicar de forma ágil de acuerdo al manifiesto ágil.

Como tercer paso, se realiza la *Integración y Mejora de las Prácticas para el Proceso de Mantenimiento*, en el cual se toman las mejores prácticas resultantes y se validan con las prácticas actuales de la organización estudiada y posteriormente se integran al proceso propuesto para la gestión del mantenimiento de acuerdo al entorno actual de la organización.

Como cuarto paso se realiza una *Propuesta de Implementación del Proceso Ágil de Gestión de Mantenimiento*, de acuerdo al proceso ágil usado actualmente por la organización estudiada

### 3.2 ORGANIZACIÓN ESTUDIADA

Se realizaron entrevistas a algunos representantes de la empresa softlogik para identificar sus procesos actuales en el ámbito de desarrollo y mantenimiento y también para validar las prácticas resultantes del estudio.

Softlogik es una empresa fundada en febrero del 2011 en la ciudad de Zacatecas con el objetivo de incorporarse al mercado local y nacional de desarrollo e implementación de soluciones basadas en procesos y tecnologías de información.

Dentro de los servicios ofrecidos por softlogik además de Desarrollo a la medida y seguridad informática, se

encuentra el SaaS, en el que involucran el desarrollo de aplicaciones, infraestructura de procesamiento y almacenamiento así como todos los aspectos inherentes al equipamiento, mantenimiento y actualización de esta infraestructura a través del servicio de “Cloud Computing”.<sup>1</sup>

Actualmente, softlogik cuenta con productos de SaaS que manejan de forma contenida a manera de proyectos piloto, sin embargo, una vez abiertos estos productos al mercado la empresa estima que se tendrá un incremento considerable en los servicios tanto de mantenimiento como de soporte en este entorno de CC; por lo que la empresa tiene interés en mejorar sus procesos de mantenimiento de tal forma que puedan ser aplicados de forma ágil y a través de las diferentes capas que integran el CC para mejorar la calidad del servicio y de esta forma impactar en la satisfacción de sus clientes.

#### 4 SELECCIÓN DE MODELOS PARA LA OBTENCIÓN DE PRÁCTICAS

Considerado que el objetivo de este estudio es definir un proceso de gestión del mantenimiento aplicado al CC, donde de acuerdo a la revisión de la literatura, el CC se puede dividir en 3 elementos principales: Software, Plataforma e Infraestructura [2], [3], [25]. Se consideran los siguientes factores para la selección de los modelos:

- 1) *La necesidad de definir un proceso de gestión de mantenimiento, y*

La aplicación del mantenimiento al CC en:

- 2) *Aplicaciones de Software para CC*
- 3) *Plataforma de operación del CC y*
- 4) *La Infraestructura o Equipamiento en el CC.*

Para el primer factor considerado, se selecciona el Modelo de Madurez de Mantenimiento de software S3M[16] en sus niveles 0,1 y 2, que están enfocadas en definir la base de la gestión del mantenimiento. Se pretende usar este modelo para establecer las bases necesarias para cumplir con la necesidad de definir el proceso de gestión de mantenimiento.

Para el segundo factor, se toma en cuenta que las aplicaciones usadas en el CC de acuerdo al modelo de servicio SaaS, se ofrecen a través del navegador como aplicaciones Web[25], considerando que el enfoque de este factor es para aplicaciones Web, se selecciona el proyecto de seguridad web OWASP[20], que es un proyecto que está enfocado completamente en la seguridad y el mantenimiento de aplicaciones web.

Para los factores 3 y 4 referentes a Plataforma e Infraestructura, se hace uso de las prácticas de la gestión de la disponibilidad que se encuentran descritas dentro del

Diseño del Servicio de ITIL v3[13], estas prácticas están enfocadas en asegurar que la infraestructura, procesos y herramientas cumplan con objetivos de disponibilidad propuestos. Adicional a esto, estas prácticas son usadas debido a que softlogik ya hace uso de ITIL para su gestión en TI. Los modelos seleccionados son descritos a continuación.

#### 4.1 S3M – MODELO DE MADUREZ DEL MANTENIMIENTO DEL SOFTWARE

El dominio del primer proceso del modelo S3M se centra en la gestión de los procesos de mantenimiento de software. El dominio de este proceso reconoce la importancia de los recursos humanos en el centro de las actividades de mantenimiento de software, a través del contacto diario con los clientes y la ejecución de las muchas actividades que integran el proceso de mantenimiento en general. S3M permite a los ingenieros de mantenimiento de software tener los procesos, técnicas y herramientas a su disposición para satisfacer a sus clientes y usuarios; además les permite tener la capacidad para ayudar en el proceso de definición, adquisición y mejora.

El modelo S3M está dividido en 5 áreas clave de la gestión del proceso de mantenimiento, estas áreas están divididas de la siguiente manera [16]:

- PRO1. Enfoque del proceso de mantenimiento.**  
(Maintenance Process Focus).
- PRO2. Definición del Servicio/Proceso.**  
(Maintenance Process/Service Definition).
- PRO3. Capacitación en mantenimiento.**  
(Maintenance Training).
- PRO4. Rendimiento del proceso de mantenimiento.**  
(Maintenance Process Performance).
- PRO5. Innovación e implementación.**  
(Maintenance Innovation and Deployment).

El objetivo del área de *enfoque del proceso de mantenimiento* es asegurar que existe la responsabilidad para recolectar la información sobre las mejoras que se necesitan, desarrollar las ideas necesarias para analizarlas, identificar soluciones de mejora basadas en prioridades, metas y restricciones.

El objetivo del área de *definición del servicio/Proceso* de mantenimiento es establecer, documentar y mantener un repositorio de procesos estándar para mantenimiento de software y la infraestructura de esos procesos. En el cual se debe desarrollar una guía para ajustar los estándares de procesos de mantenimiento de software.

El objetivo del área de *capacitación en mantenimiento* de software es identificar las necesidades de capacitación necesarias y planes de capacitación; además, se identifican los perfiles del personal que se requiere para el nuevo software. En esta área se incluyen los planes de capacitación al personal y a los usuarios.

<sup>1</sup> Softlogik, Información Consultada en su página Web el 11 de Marzo de 2013. [www.softlogik.mx](http://www.softlogik.mx)

El objetivo del área de *Rendimiento del proceso de mantenimiento* es obtener y mantener una descripción cuantitativa del rendimiento de las actividades clave del proceso de mantenimiento de software. Esto ayudará a explicar el actual rendimiento del proceso/servicio y establecer los objetivos/metás de calidad.

En el área de *Innovación e implementación* del mantenimiento se identifican e implementan mejoras e innovaciones que tendrán un impacto significativo (y que sea medible) en el rendimiento, calidad y rentabilidad del proceso/servicio de mantenimiento de software y sus tecnologías.

#### 4.2 ITIL – V3 DISEÑO DEL SERVICIO

ITIL [13] es un conjunto de 5 áreas de prácticas enfocadas en gestión de los servicios de las TI. Estas 5 áreas están divididas en: Estrategia del servicio, Diseño del servicio, Transición del servicio, Operación del servicio y Mejora continua del servicio. Esta revisión se enfoca en la gestión de la disponibilidad que está comprendida dentro de la segunda área de prácticas que corresponde al diseño del Servicio.

El área del diseño de servicio se enfoca en proporcionar una guía para el diseño y desarrollo de los servicios y procesos de gestión de los servicios, además de que su alcance no se limita a los nuevos servicios, si no que incluye los cambios y mejoras necesarias para aumentar o mantener el valor a los clientes sobre los servicios durante su ciclo de vida, su continuidad, el logro de niveles de servicio y la conformidad con las normas y reglamentos.

Dentro del **diseño del servicio** se manejan procedimientos que son descritos generalmente en la guía práctica de Gestión del servicio de ITIL, estos procedimientos se dividen en:

- Gestión del catálogo de Servicios.
- Gestión del nivel de servicio.
- Gestión de la capacidad.
- **Gestión de la disponibilidad.**
- Gestión de la continuidad de los servicios TI.
- Gestión de la seguridad de la información.
- Gestión de proveedores.

Donde todos estos procedimientos deben estar vinculados a la gestión, el diseño, el soporte y el mantenimiento de los servicios, la infraestructura, el medio ambiente, a las aplicaciones y a los datos, lo que nos permite cubrir las diferentes capas de abstracción del CC.

El objetivo de la gestión de la Disponibilidad es definir, analizar, planificar, medir y mejorar la disponibilidad de los servicios de TI en todos los aspectos. La gestión de la disponibilidad se encarga de asegurar que la **infraestructura**, los procesos, las herramientas y las funciones de TI sean adecuados para cumplir con los objetivos de disponibilidad propuestos.

#### 4.3 OWASP (PROYECTO ABIERTO DE SEGURIDAD DE APLICACIONES WEB) - GUÍA DE DESARROLLO

OWASP [28] es un espacio abierto a la comunidad dedicada a la búsqueda y la lucha contra las causas del software inseguro. La guía de desarrollo de OWASP proporciona orientación detallada sobre la seguridad en aplicaciones web. En esta guía se presentan detalles sobre la seguridad de la mayoría de las formas de aplicaciones web y servicios.

“La seguridad no es un terreno blanco o negro, sino que son muchos tonos de gris”, se menciona en la guía de desarrollo para indicarnos que la seguridad no se puede manejar como algo que se pueda cubrir completamente, sino que más bien, se tienen que buscar los recursos y las posibilidades de que las aplicaciones cubran a mayor medida los riesgos y las amenazas de seguridad.

La guía de desarrollo de OWASP, cubre diferentes aspectos de la seguridad web, dentro de los cuales integra desde principios de codificación segura, modelado de riesgos y amenazas hasta prácticas específicas en diferentes elementos que integran los proyectos web, como Manejo de pagos en comercio electrónico, Phising (robo de información confidencial), Servicios web, Autenticación, Autorización, Manejo de sesiones, Validación de datos, Interpretes de inyección, Manejo de errores, auditorías, sistemas de ficheros, desbordamientos de memoria, interfaces administrativas, cifrados, comunicación, mantenimiento y ataques de denegación de servicio.

#### 5 IDENTIFICACIÓN Y REDUCCIÓN DE PRÁCTICAS PARA DISPONIBILIDAD Y MÉTODOS LEST

Tomando como base S3M, OWASP e ITIL-v3, se analizaron y se identificaron sus prácticas principales para aplicar en el proceso de gestión de mantenimiento. Todas estas prácticas fueron analizadas de tal forma que se les aplicaron tres filtros específicos para reducir el número total de prácticas buscando ajustarlas al proceso de gestión de mantenimiento y lograr el aseguramiento del atributo de calidad de disponibilidad y su alineación a los principios ágiles.

La reducción de prácticas, se muestra en la **Fig. 1**, que consiste en aplicar 3 filtros específicos, en el primer filtro se extraen las prácticas de cada uno de los modelos conforme a actividades que puedan ser aplicadas en un proceso de gestión de mantenimiento, (El modelo S3M no es filtrado y se pasa completo al siguiente filtro porque este modelo es usado como criterio de filtro). El segundo filtro, consiste en extraer las mejores prácticas resultantes del filtro anterior alineándolas a aquellas prácticas que permitan asegurar la disponibilidad de los servicios en entornos CC. En el tercer filtro, se extraen de las mejores prácticas resultantes del segundo filtro aquellas que se puedan alinear al manifiesto ágil de desarrollo de software. El resultado de estas actividades es un listado de prácticas que ya pueden ser usadas en la gestión del mantenimiento y están alineadas a la disponibilidad y al mani-

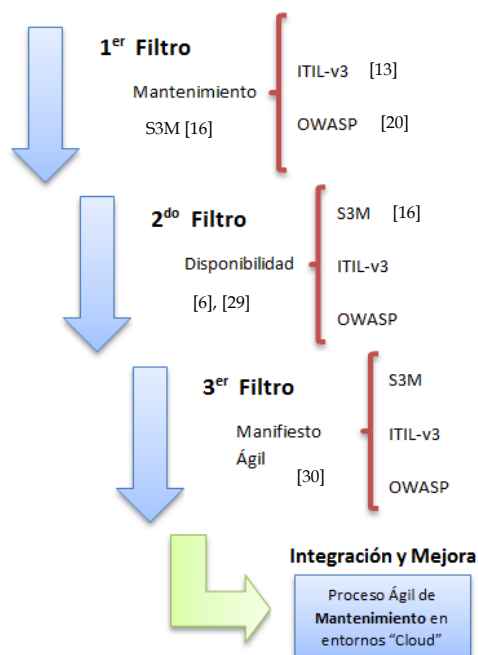


Fig. 1 Aplicación de filtros en la reducción de Prácticas.

fiesto ágil.

El seguimiento de la aplicación de todos los filtros, se puede consultar en el ANEXO 1 donde se representan las prácticas que fueron seleccionadas y descartadas en cada uno de los filtros además del principio ágil que se relaciona a las prácticas finales y el tipo de capa del CC al que se pretenden aplicar.

## 5.1 PRIMER FILTRO (MANTENIMIENTO)

Para realizar la reducción de las prácticas en este primer filtro, se identificaron aquellas prácticas que pudieran ser alineadas a un proceso de gestión de mantenimiento. Este primer filtro, está enfocado en el proceso de gestión del mantenimiento de S3M (Por lo que S3M no entra en este filtro y pasa directo al segundo filtro).

En S3M el mantenimiento se enfoca en la *gestión del proceso de mantenimiento de software* donde se deben satisfacer los criterios esperados por los clientes y usuarios por medio del contacto diario y por la ejecución de sus tareas dentro del proceso de mantenimiento; este proceso en S3M está enfocado en cubrir 5 áreas específicas que son: el enfoque en el proceso, la definición del servicio y procesos, capacitaciones, rendimiento del proceso y la innovación e implementación; todo esto dentro del mantenimiento.

Considerando estas 5 áreas de S3M, se revisaron las prácticas de ITIL y de OWASP y se seleccionaron aquellas que tuvieran:

- Relación con alguna de las 5 áreas de proceso y
- que pudieran ser aplicadas a la gestión de las actividades de mantenimiento considerando que sean acti-

vidades que implicaran hacer cambios relacionados al software o infraestructura después de la entrega o cuando ya están en funcionamiento.

Las prácticas resultantes de la aplicación de este filtro se muestran en la TABLA 1.

## 5.2 SEGUNDO FILTRO (DISPONIBILIDAD)

Considerando las prácticas resultantes del primer filtro, se les aplicó este segundo filtro para continuar con la reducción, donde se identificaron todas aquellas prácticas que pudieran ser alineadas para asegurar el atributo de calidad de disponibilidad del servicio en entornos CC.

La realización de este filtro estuvo basada en las definiciones de disponibilidad presentadas en:

- El reporte técnico CMU/SEI-95-TR-021 [29] del Software Engineering Institute (SEI) y
- En la definición de disponibilidad en entornos de CC presentado por Da-Wei Sun et al. [6] en su artículo.

El SEI divide la disponibilidad conforme a 2 criterios, la **confiabilidad** y la **seguridad**; respecto a la confiabilidad se describe la disponibilidad como *la medida de que tan listo está el sistema para ser usado*; y con respecto a la seguridad la define como *la propiedad que deben tener los sistemas para que los recursos que maneja estén realmente disponibles para el usuario autorizado*.

Da-Wei Sun menciona que en el CC, la disponibilidad puede referirse al *tiempo de vida de un sistema* y que puede ser dividido en un conjunto de estados de tipo "up states" y de "down states", en el que la disponibilidad puede ser categorizada como disponibilidad instantánea, disponibilidad estable y disponibilidad inherente.

**La disponibilidad instantánea** la definen como la probabilidad que tiene un sistema de que esté en "up state" al iniciar la solicitud del servicio, por lo que el sistema debiera estar disponible al momento en que se solicita.

**La disponibilidad estable** la definen como la probabilidad que tiene un sistema de estar en "up state" por un periodo de tiempo que sea lo suficientemente largo después de que se inicia y es examinado o utilizado.

**La disponibilidad inherente** de un sistema es el porcentaje esperado en intervalo de tiempo durante el cual el sistema realiza las funciones requeridas [6].

Considerando estas definiciones se seleccionaron aquellas prácticas de los modelos que de una u otra forma estuvieran involucradas o fueran viables de aplicarse enfocadas en el **control, medición y mantenimiento** de:

- Los 2 estados ("up state" y "Down state") que se presentan en el servicio de CC y
- En la disponibilidad conforme a seguridad y confia-

bilidad.

En la TABLA 2, se pueden apreciar las prácticas resultantes de aplicar este segundo filtro.

### 5.3 TERCER FILTRO (MANIFIESTO ÁGIL)

Considerando las prácticas resultantes del segundo filtro presentes en la TABLA 2, se aplica un tercer filtro para reducir estas prácticas, para lo cual se toman como base los 12 principios del manifiesto ágil de desarrollo de software propuesto por Kent Beck et al. [30] para establecer la agilidad de las prácticas seleccionadas para el proceso, estos principios son los siguientes:

1. Prioridad en la satisfacción del cliente mediante entregas tempranas y continuas de software con valor.
2. Se aceptan los cambios en etapas tardías del desarrollo.
3. Entrega de funcionalidad en periodos de entre 2 semanas a 2 meses o antes.
4. Los responsables del negocio y desarrolladores trabajan de forma conjunta.
5. Los proyectos se realizan con gente motivada.
6. El método más eficiente de comunicación entre el equipo de desarrollo es la conversación cara a cara.
7. Software funcionando es la principal métrica de progreso.
8. El equipo debe de mantener un ritmo de trabajo constante, basado en la agilidad como un desarrollo sustentable.
9. La atención continua a la excelencia técnica y el buen diseño mejora la Agilidad.
10. Mantener la simplicidad – El arte de optimizar la cantidad de trabajo no hecho.
11. Las mejores arquitecturas, requerimientos y diseños surgen de equipos auto-organizados.
12. Tener periodos de reflexión de cómo puede ser

**TABLA 1 REDUCCIÓN DE PRÁCTICAS ITIL Y OWASP (FILTRO 1: MANTENIMIENTO)**

**FILTRO 1: MANTENIMIENTO**

Área		#	Práctica			
ITIL-v3 Diseño del Servicio	Gestión de la Disponibilidad	Actividades Reactivas	1 Monitorear, medir, analizar y reportar la disponibilidad de los componentes y el servicio.			
			2 Análisis de ausencia de disponibilidad.			
			4 Análisis de fallas en el Servicio.			
			5 Identificar las funciones vitales del negocio o VBFs (Vital Business Functions).			
			6 Diseñara para disponibilidad.			
		Actividades Proactivas	7 Análisis de impacto de fallas en los componentes.			
			8 Análisis de punto único de fallo o SPoF (Single Point of Failure).			
			10 Modelado.			
			11 Gestión y análisis de Riesgos			
			12 Programación de pruebas de disponibilidad.			
			13 Mantenimiento planificado y preventivo.			
			15 Revisiones continuas y mejoras.			
			OWASP – Guía de desarrollo	Manejo de Sesiones	6 Considerar cuidadosamente donde guarda el estado de la aplicación	
					Manejo de errores, auditoria y Logs	9 Prueba de fallas – no deje sin gestionar las fallas.
						10 Utilice logs de doble propósito
11 Los logs de auditoria se encuentran legalmente protegidos – ¡protéjalos!						
12 Genere reportes y realice búsquedas en los logs utilizando una copia de solo lectura o una copia íntegra del original.						
Sistema de Ficheros	13 Usar jaulas “chroot” en plataformas Unix.					
	14 Usar los mínimos permisos en el sistema de ficheros de todas las plataformas.					
Mantenimiento	Interfases Administrativas	15 Considerar la utilización de sistemas de fichero de solo lectura (como CD-ROM o llaves USB bloqueadas) si fuera posible.				
		19 Diseñar cuidadosamente la funcionalidad de servicio de asistencia / moderador / soporte al cliente alrededor de una capacidad administrativa limitada y aplicación segregada o acceso.				
	25 Es responsabilidad como desarrollador garantizar que el usuario este tan seguro como sea posible y animar a parchear rápidamente los sistemas vulnerables para garantizar que sus parches sean integrales (es decir, no más parches de este tipo), regresión de problemas anteriores (es decir, soluciones permanecen solventadas), y estable (es decir, que se han realizado suficientes ensayos).					
	26 Las aplicaciones deben mantenerse con regularidad, en busca de nuevos métodos para evitar los controles de seguridad					
	27 Es normal dentro de la industria proporcionar apoyo desde n-1 hasta n-2 versiones, por lo que se requiere algún tipo revisión de control de código fuente, como CVS, ClearCase, o subversión, son necesarios para gestionar las correcciones de seguridad y evitar errores de regresión.					
	28 Las actualizaciones deben ser prestadas de forma segura, ya sea por firma digital de los paquetes, o mediante un resumen de mensaje libre de colisiones.					
	29 El apoyo a la política de parches de seguridad deben ser claramente comunicada a los usuarios, para asegurar que los usuarios concocen que versiones están soportadas para soluciones de seguridad y cuando termina su vida.					

más eficientes y entonces ajustar y perfeccionar su comportamiento.

Las prácticas seleccionadas en este filtro, fueron aquellas que se consideró pueden ser implementadas dentro

de un proceso ágil cumpliendo con al menos uno de estos principios. A continuación en la TABLA 3 se presentan las prácticas resultantes al aplicar el manifiesto ágil en este tercer filtro.

**TABLA 2 REDUCCIÓN DE PRÁCTICAS S3M, ITIL Y OWASP (FILTRRO 2: DISPONIBILIDAD)  
(PARTE 1 DE 2)**

**FILTRO 2: DISPONIBILIDAD**

Área <sup>2</sup>	#	Práctica
S3M	PRO1	1 Un programa de mejora de la calidad y del proceso ha sido iniciado en el departamento de TI a nivel organizacional. Los gerentes del área de mantenimiento de software están enterados de este programa y han recibido un entrenamiento de concientización inicial sobre la calidad.
		2 Los resultados de encuestas de mantenimiento de software aplicados a los clientes sobre los productos/servicios son usados para identificar mejoras.
		3 Las observaciones, comentarios y quejas de los usuarios/clientes y los grupos de interfaces son recolectados y usados para identificar mejoras.
		4 Los datos de las fallas del software son recolectados y usados para identificar mejoras al proceso de mantenimiento/producto y también a las interfaces con los otros grupos de interfaces.
		5 Se realiza una lista con las mejoras candidatas a implementarse ordenadas por prioridades, y son usadas como guía para desarrollar el programa de mejora del mantenimiento de software.
		6 Las 10 posibles mejoras de la lista son discutidas, se realizan planes de acción por los medios y altos directivos. Las actividades/proyectos planeadas son soportadas dentro del presupuesto anual de mantenimiento y operaciones actuales.
	PRO2	7 Exista al menos una descripción del proceso de mantenimiento/servicio que es usado por el cliente y es ofrecido por la organización.
		8 Partes del proceso de mantenimiento/servicios/recursos son documentados e implementados en la organización.
	PRO3	9 El ingeniero de mantenimiento de software es entrenado y motivado a realizar bien las cosas cuando usa el proceso/servicio y ejecuta su rol.
		10 Existen planes y registros que describen el nivel educativo y entrenamiento necesario para cada puesto de ingeniero de mantenimiento y de aplicaciones de software. Este plan de entrenamiento incluye capacitaciones necesarias, cursos ofrecidos, y material relacionado, créditos, recursos disponibles y el plan educativo y actividades de entrenamiento.
		11 El tiempo de capacitación es planeado.
	PRO4	12 Algunos procesos de mantenimiento clave, servicios y productos son medidos. Estos son definidos y usados localmente.
		13 Líneas base de medida de la calidad y rendimiento del proceso, servicios, y productos son recolectados, almacenados, revisados y usados con los stakeholders. Estos datos son usados para mejorar el proceso actual, los servicios y/o productos.
		14 La organización de mantenimiento ha establecido algunos objetivos de rendimiento y calidad.
	PRO5	15 Nuevos procesos, servicios, tecnologías, metodologías y herramientas son identificados e investigados por su potencial uso en el mantenimiento de software.
		16 Nuevos procesos, servicios, tecnologías, metodologías y herramientas son alcanzados e introducidos en el nivel requerido.
		17 Mejoras a algunos procesos de mantenimiento, servicios, tecnologías, metodologías y herramientas han sido iniciados de una forma controlada localmente.
ITIL-v3 Diseño del servicio	Actividades Reactivas	1 Monitorear, medir, analizar y reportar la disponibilidad de los componentes y el servicio.
		2 Análisis de ausencia de disponibilidad.
		4 Análisis de fallas en el Servicio.
		5 Identificar las funciones vitales del negocio o VBFs (Vital Business Functions).
		6 Diseñara para disponibilidad.
	Actividades Proactivas	7 Análisis de impacto de fallas en los componentes.
		8 Análisis de punto único de fallo o SPoF (Single Point of Failure).
		10 Modelado.
		11 Gestión y análisis de Riesgos
		12 Programación de pruebas de disponibilidad.
		13 Mantenimiento planificado y preventivo.
		15 Revisiones continuas y mejoras.

<sup>1</sup> Área de gestión del proceso (PRO) para las mejores prácticas de los niveles 0, 1 y 2 en S3M.

**TABLA 2 REDUCCIÓN DE PRÁCTICAS S3M, ITIL Y OWASP (FILTRO 2: DISPONIBILIDAD) (PARTE 2 DE 2)**

**FILTRO 2: DISPONIBILIDAD**

Área	#	Práctica
OWASP – Guía de Desarrollo	6	Manejo de Sesiones Considerar cuidadosamente donde guarda el estado de la aplicación
	9	Manejo de errores, auditoría y Logs Prueba de fallas – no deje sin gestionar las fallas.
	10	Utilice logs de doble propósito
	12	Genere reportes y realice búsquedas en los logs utilizando una copia de solo lectura o una copia íntegra del original.
	13	Usar jaulas “chroot” en plataformas Unix.
	14	Sistema de Archivos Usar los mínimos permisos en el sistema de ficheros de todas las plataformas.
	15	Considerar la utilización de sistemas de fichero de solo lectura (como CD-ROM o llaves USB bloqueadas) si fuera posible.
	19	Interfaces Administrativas Diseñar cuidadosamente la funcionalidad de servicio de asistencia / moderador / soporte al cliente alrededor de una capacidad administrativa limitada y aplicación segregada o acceso.
	25	Mantenimiento Es responsabilidad como desarrollador garantizar que el usuario este tan seguro como sea posible y animar a parchear rápidamente los sistemas vulnerables para garantizar que sus parches sean integrales (es decir, no más parches de este tipo), regresión de problemas anteriores (es decir, soluciones permanecen solventadas), y estable (es decir, que se han realizado suficientes ensayos).
	26	Las aplicaciones deben mantenerse con regularidad, en busca de nuevos métodos para evitar los controles de seguridad
	28	Las actualizaciones deben ser prestadas de forma segura, ya sea por firma digital de los paquetes, o mediante un resumen de mensaje libre de colisiones.

**6 INTEGRACIÓN Y MEJORA DE LAS PRÁCTICAS PARA EL PROCESO DE MANTENIMIENTO**

Las prácticas resultantes de la aplicación de los filtros nos generan un listado de prácticas que ya pueden ser alineadas a un proceso de mantenimiento, pueden ser enfocadas en satisfacer la disponibilidad de los servicios y también pueden ser tratadas dentro del marco de las metodologías ágiles, sin embargo, este listado de prácticas aun esta alienada al modelo al cual pertenece, por lo que por sí solas, es difícil que generen valor, debido a esto se lleva a cabo un proceso de integración y mejora de estas prácticas para enfocarnos en cubrir un proceso ágil en gestión de mantenimiento para entornos CC en el cual pueda ser usado en las diferentes capas de abstracción de estos entornos (Infraestructura, Plataforma y Software).

En la Fig. 2 se presentan las actividades realizadas en

este proceso de integración y mejora en las que se llevan a cabo las actividades de: Mejores Prácticas + Enfoque para enfocar las prácticas a cubrir los problemas en disponibilidad, la Validación de las prácticas con la organización, un Resumen de prácticas para su fácil manejo, la Clasificación y adaptación de las prácticas para ser manejadas en la organización y la selección de herramientas de soporte para las actividades establecidas por las prácticas resultantes. Todas estas actividades y sus resultados, son descritos a continuación.

Los resultados finales obtenidos a través del proceso de integración y mejora de las prácticas realizado y descrito a través de los puntos 0-6.6 de este documento son descritos y modelados en el ANEXO 4 usando la herramienta *Bizagi Process Modeler* para permitir la visualización de la secuencia y conexión entre cada una de las prácticas, los enfoques y las herramientas definidas.

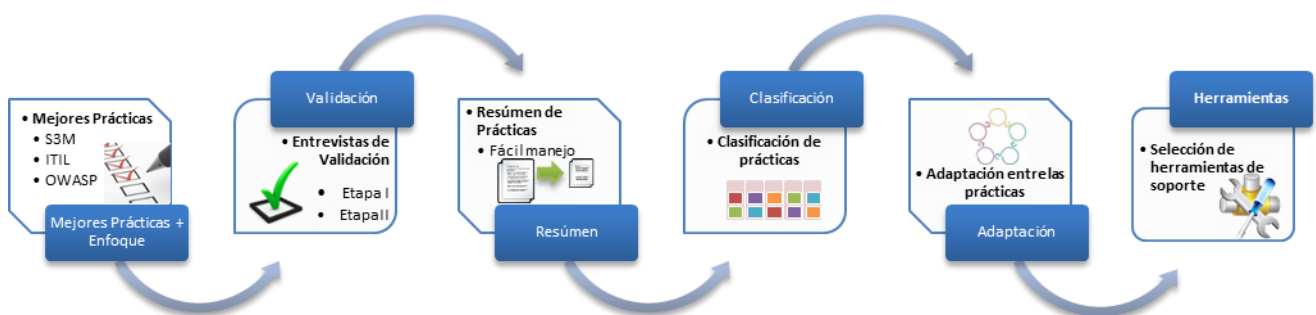


Fig. 2 Proceso de integración y mejora de las prácticas para el proceso de mantenimiento.

**TABLA 3** REDUCCIÓN DE PRÁCTICAS S3M, ITIL Y OWASP (FILTRO 3: MANIFIESTO ÁGIL)**FILTRO 3: MANIFIESTO ÁGIL**

Área	#	Práctica	ID <sup>1</sup>
S3M	PRO1	1 Un programa de mejora de la calidad y del proceso ha sido iniciado en el departamento de TI a nivel organizacional. Los gerentes del área de mantenimiento de software están enterados de este programa y han recibido un entrenamiento de concientización inicial sobre la calidad.	1
		2 Los resultados de encuestas de mantenimiento de software aplicados a los clientes sobre los productos/servicios son usados para identificar mejoras.	2
		3 Las observaciones, comentarios y quejas de los usuarios/clientes y los grupos de interfaces son recolectados y usados para identificar mejoras.	3
		5 Se realiza una lista con las mejoras candidatas a implementarse ordenadas por prioridades, y son usadas como guía para desarrollar el programa de mejora del mantenimiento de software.	4
	PRO2	7 Exista al menos una descripción del proceso de mantenimiento/servicio que es usado por el cliente y es ofrecido por la organización.	5
	PRO3	9 El ingeniero de mantenimiento de software es entrenado y motivado a realizar bien las cosas cuando usa el proceso/servicio y ejecuta su rol.	6
		11 El tiempo de capacitación es planeado.	7
	PRO4	12 Algunos procesos de mantenimiento clave, servicios y productos son medidos. Estos son definidos y usados localmente.	8
		14 La organización de mantenimiento ha establecido algunos objetivos de rendimiento y calidad.	9
	PRO5	15 Nuevos procesos, servicios, tecnologías, metodologías y herramientas son identificados e investigados por su potencial uso en el mantenimiento de software.	10
		16 Nuevos procesos, servicios, tecnologías, metodologías y herramientas son alcanzados e introducidos en el nivel requerido.	11
		17 Mejoras a algunos procesos de mantenimiento, servicios, tecnologías, metodologías y herramientas han sido iniciados de una forma controlada localmente.	12
ITIL-v3 Diseño del servicio	Actividades Reactivas	2 Análisis de ausencia de disponibilidad.	13
		Actividades Proactivas	5 Identificar las funciones vitales del negocio o VBFs (Vital Business Functions).
	8 Análisis de punto único de fallo o SPoF (Single Point of Failure).		15
	12 Programación de pruebas de disponibilidad.		16
	13 Mantenimiento planificado y preventivo.		17
	15 Revisiones continuas y mejoras.	18	
OWASP – Guía de desarrollo	Manejo de errores, auditoria y Logs	9 Prueba de fallas – no deje sin gestionar las fallas.	19
		12 Genere reportes y realice búsquedas en los logs utilizando una copia de solo lectura o una copia íntegra del original.	20
	Sistema de Ficheros	13 Usar jaulas “chroot” en plataformas Unix.	21
		14 Usar los mínimos permisos en el sistema de ficheros de todas las plataformas.	22
		15 Considerar la utilización de sistemas de fichero de solo lectura (como CD-ROM o llaves USB bloqueadas) si fuera posible.	23
	Interfaces Administrativas	19 Diseñar cuidadosamente la funcionalidad de servicio de asistencia / moderador / soporte al cliente alrededor de una capacidad administrativa limitada y aplicación segregada o acceso.	24
	Mantenimiento	25 Es responsabilidad como desarrollador garantizar que el usuario este tan seguro como sea posible y animar a parchear rápidamente los sistemas vulnerables para garantizar que sus parches sean integrales (es decir, no más parches de este tipo), regresión de problemas anteriores (es decir, soluciones permanentes solventadas), y estable (es decir, que se han realizado suficientes ensayos).	25
		26 Las aplicaciones deben mantenerse con regularidad, en busca de nuevos métodos para evitar los controles de seguridad	26
		28 Las actualizaciones deben ser prestadas de forma segura, ya sea por firma digital de los paquetes, o mediante un resumen de mensaje libre de colisiones.	27

<sup>1</sup> ID, identificador para permitir la rastreabilidad a lo largo de los análisis posteriores realizados a estas prácticas.



## 6.1 MEJORES PRÁCTICAS + ENFOQUE

En esta actividad se tomaron las prácticas resultantes de aplicar los filtros y se enfocaron en satisfacer los principales problemas detectados en la búsqueda la disponibilidad del servicio.

De acuerdo a la naturaleza de cada una de las prácticas, estas están ya enfocadas en satisfacer las necesidades del modelo al cual pertenecen (S3M para gestión de mantenimiento, gestión de la disponibilidad de ITIL para la disponibilidad del servicio y OWASP para la seguridad en las aplicaciones web), sin embargo, gracias a los filtros aplicados, estas prácticas ya podrían ser utilizadas para establecer un proceso ágil en la gestión del mantenimiento que sirva como base para la organización y que pueda satisfacer el atributo de calidad de disponibilidad en el servicio. Para lograr esto, se considera agregar a las prácticas obtenidas el *enfoque* (+Enfoque) requerido para satisfacer los problemas de tolerancia a fallos, limitación de recursos y replicación de datos. Para facilitar esta asignación de enfoques, se identifica la relación de cada uno de los problemas con cada una de las prácticas. Las prácticas del modelo S3M, al estar enfocadas en la gestión de un proceso de mantenimiento a nivel general, se consideró que fácilmente pueden relacionarse con los 3 problemas. Las prácticas de los modelos OWASP e ITIL se relacionaron de acuerdo a la definición de cada una de sus prácticas; esta relación se muestra en la columna *Problema* de la TABLA 13 en el ANEXO 2.

Al tomar estas Mejores Prácticas y agregarles un enfoque específico en satisfacer los problemas de la disponibilidad en tolerancia a fallos, limitación de recursos y replicación de datos, se pretende que nos permita dejar las prácticas base de forma genérica para posteriormente **poder asignar otro enfoque** en caso de que se requiera abordar otro tipo de problemas de disponibilidad que no están siendo considerados en este estudio.

En la TABLA 4 se muestran los enfoques aplicados a algunas de las prácticas, otras prácticas, no se consideraron para integrarles enfoques debido a que pueden ser implementadas de la forma en que están actualmente especificadas. En esta tabla las prácticas son referenciadas por el ID (identificador) que le corresponde de acuerdo a la TABLA 3 de las prácticas resultantes de los filtros.

## 6.2 VALIDACIÓN DE LAS PRÁCTICAS RESULTANTES

En esta actividad, se realizaron entrevistas para validar las prácticas resultantes con respecto al trabajo actual de softlogik. Las entrevistas se hicieron en 2 etapas, en la primera etapa se abarcaron las prácticas referentes al modelo S3M y en la segunda etapa se abarcaron las prácticas de los modelos OWASP e ITIL. La validación de las prácticas se hizo en el sentido de identificar si las prácticas ya eran implementadas por la organización y en caso de que si las llevaran a cabo identificar desde *cuando* y *como* son implementadas y si les brinda algún *beneficio* para la organización.

**TABLA 4 ENFOQUES DE LAS PRÁCTICAS**

ID	+Enfoque
1	Concientización inicial sobre el uso de un plan de mantenimiento y calidad de la disponibilidad de los servicios atacando los problemas de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos.
2	Las encuestas enfocadas en identificar mejoras en el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos.
3,4	Identificar mejoras en el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos
6	Capacitación en el área de disponibilidad del servicio cubriendo principalmente el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos
9	Enfocados en mejorar la disponibilidad del servicio cubriendo el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos
10	Investigación en el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos
11	Nuevos elementos a integrar enfocadas en el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos para el aseguramiento de la disponibilidad.
12	Mejoras en el área de (1) tolerancia a fallas, (2) Limitación de recursos y (3) Replicación de datos para el aseguramiento de la disponibilidad.
17	Se hace planificación del mantenimiento para prevenir la falta de la disponibilidad.
21	Aplicar cuando se considere necesario de acuerdo a los requerimientos del proyecto.
26	Mantenerse con regularidad para identificar nuevas amenazas que afecten la disponibilidad del servicio.

### 6.2.1 ETAPA 1

La entrevista que se realiza en esta primera etapa aborda las prácticas del modelo S3M y es aplicada a 2 miembros de la organización, el Director de Operaciones y un Ingeniero Sénior, comprendiendo el nivel directivo y operativo de la organización. los resultados de esta entrevista se muestran en la primera sección de la TABLA 13 incluida en el ANEXO 2 que comprende las prácticas de la 1 a la 12, en esta tabla se puede apreciar que de las 12 prácticas del modelo solamente 3 son implementadas completamente, 4 se implementan parcialmente y 5 no son consideradas por la organización.

### 6.2.2 ETAPA 2

En la entrevista de esta segunda etapa se abordaron las prácticas de los modelos ITIL y OWASP, la entrevista se aplicó solamente al Director de Operaciones cubriendo solo el nivel directivo. Los resultados de esta entrevista se muestran en la segunda sección de la TABLA 13 incluida en el ANEXO 2 que comprende las prácticas de la 13 a la 27, en esta tabla se puede apreciar que de las 15 prácticas de los modelos 7 son implementadas completamente, 4 se implementan parcialmente y 4 no son consideradas por la organización.

## 6.3 RESUMEN DE PRÁCTICAS

Debido a la longitud de algunas de las definiciones de las prácticas utilizadas en este estudio, se define un término corto para cada una de ellas para facilitar su manejo y

representación dentro de los procesos. En la TABLA 5 se muestra el resumen de las prácticas.

**TABLA 5 TÉRMINOS CORTOS DE LAS PRÁCTICAS**

ID	Término corto de la Práctica
1	Iniciar plan de mejora de calidad y de proceso de mantenimiento
2	Aplicar encuestas
3	Usar las observaciones, comentarios y quejas para identificar mejoras
4	Listar y priorizar las mejoras identificadas
5	Definir la descripción del proceso de mantenimiento/servicio
6	Planificar capacitaciones referentes al proceso de mantenimiento
7	Planear los tiempos de capacitación
8	Identificar procesos de mantenimiento clave y aplicarles métricas
9	Establecer objetivos de rendimiento y calidad
10	Identificar e investigar Novedades y mejoras potenciales en el uso del proceso de mantenimiento.
11	Integrar los nuevos hallazgos de las investigaciones en el nivel requerido.
12	Las mejoras a los procesos actuales, son iniciados de forma contenida localmente
13	Realizar análisis de ausencia de disponibilidad
14	Identificar las VBF's
15	Realizar análisis de SPoF
16	Planear e Integrar al proceso pruebas de disponibilidad.
17	Planificar el Mantenimiento preventivo.
18	Revisiones y mejoras continuas.
19	Prueba de fallas - gestión de fallas
20	Usar los logs (con copias de solo lectura) para identificar problemas y buscar mejoras.
21	Configurar jaulas "chroot" en plataformas Unix
22	Configurar el mínimo de permisos en sistema de ficheros en todas las plataformas
23	Si es posible, usar sistemas de fichero de solo lectura
24	Diseñar y establecer el funcionamiento del servicio de asistencia/moderador/soporte al cliente
25	Parchear rápidamente los sistemas.
26	Planificar periodos de revisión en busca de nuevos métodos o funcionamiento.
27	Realizar actualizaciones de forma segura.

#### 6.4 CLASIFICACIÓN DE PRÁCTICAS

En esta actividad, se realiza una clasificación de las prácticas de acuerdo a las 5 áreas de proceso (PRO) establecidas por S3M para el mantenimiento.

El modelo S3M está dividido en 5 áreas clave para la gestión del proceso de mantenimiento (AP) las cuales son descritas en el punto 4.1 de este documento, las áreas consideradas en base a estas áreas son: (AP1) Enfoque del proceso de mantenimiento, (AP2) Definición del Proceso/Servicio, (AP3) Capacitación en mantenimiento, (AP4) Rendimiento del proceso/producto/servicio de mantenimiento, (AP5) Innovación, mejora e implementación.

Se decide usar las áreas de proceso de S3M debido a que están enfocadas en áreas de mantenimiento que ya han sido probadas por el modelo, además de que consi-

derando que se están usando solo los niveles 0, 1 y 2 de este modelo, el usar esta clasificación permitiría integrar fácilmente los niveles faltantes en caso de requerirse. Por su parte las prácticas de ITIL y OWASP seleccionadas pueden adaptarse fácilmente dentro de estas áreas de proceso ya que durante el primer filtro, estas fueron algunas de las condiciones para seleccionarlas. Las prácticas de la gestión de la disponibilidad de ITIL y las prácticas de OWASP pueden ser incluidas dentro de las áreas 1, 2, 4 y 5 por que cuentan con prácticas enfocadas en definir o establecer el proceso, medir el rendimiento del proceso y del producto además de que también cuentan con prácticas enfocadas en la mejora continua. Las prácticas clasificadas de acuerdo a estas áreas de proceso establecidas se muestran a continuación en la TABLA 6 .

**TABLA 6 CLASIFICACIÓN DE LAS PRÁCTICAS**

ID	Práctica	AP <sup>1</sup>
1	Iniciar plan de mejora de calidad y de proceso de mantenimiento	1
2	Aplicar encuestas	1
3	Usar las observaciones, comentarios y quejas para identificar mejoras	1
4	Listar y priorizar las mejoras identificadas	1
14	Identificar las VBFs	1
24	Diseñar y establecer el funcionamiento del servicio de asistencia/moderador/soporte al cliente	1
5	Definir la descripción del proceso de mantenimiento/servicio	2
16	Planear e Integrar al proceso pruebas de disponibilidad.	2
17	Planificar el Mantenimiento preventivo.	2
26	Planificar periodos de revisión en busca de nuevos métodos o funcionamiento.	2
6	Planificar capacitaciones referentes al proceso de mantenimiento	3
7	Planear los tiempos de capacitación	3
8	Identificar procesos de mantenimiento clave y aplicarles métricas	4
9	Establecer objetivos de rendimiento y calidad	4
13	Realizar análisis de ausencia de disponibilidad	4
15	Realizar análisis de SPoF	4
19	Prueba de fallas - gestión de fallas	4
21	Configurar jaulas "chroot" en plataformas Unix	4
22	Configurar el mínimo de permisos en sistema de ficheros en todas las plataformas	4
23	Si es posible, usar sistemas de fichero de solo lectura	4
25	Parchear rápidamente los sistemas.	4
27	Realizar actualizaciones de forma segura	4
10	Identificar e investigar Novedades* y mejoras* potenciales en el uso del proceso de mantenimiento.	5
11	Integrar los nuevos hallazgos* de las investigaciones en el nivel requerido.	5
12	Las mejoras a los procesos actuales, son iniciados de forma contenida localmente	5
18	Revisiones y mejoras continuas.	5
20	Usar los logs (con copias de solo lectura) para identificar problemas y buscar mejoras.	5

<sup>1</sup> AP, Área de proceso de mantenimiento.

### 6.5 ADAPTACIÓN DE PRÁCTICAS

En esta actividad se realiza la adaptación de las prácticas para permitir la correcta definición e interacción entre ellas además de reducir el número de prácticas finales debido a que algunas están enfocadas en un mismo tipo de actividad.

En la TABLA 7 se muestra como las prácticas fueron agrupadas y definidas en una secuencia para integrarse y darle sentido a su realización dentro de las tareas de la gestión de mantenimiento. También, las prácticas 10 y 18 fueron combinadas en una sola práctica ya que ambas se enfocan en integrar revisiones y mejoras continuas; esta nueva práctica se dejó con el número de práctica 10. Las prácticas 16 y 20 fueron integradas como tareas de soporte opcionales a sus prácticas superiores 17 y 10 respectivamente. Al realizar esta adaptación, se identificaron también grupos de prácticas específicas en tratar el performance y la configuración tanto del proceso como de los productos y servicios, por lo que se definieron 2 nuevas actividades para agruparlas. Estas actividades se definieron como *análisis de rendimiento* y *análisis de configuración* integrando como sub actividades las prácticas 13, 15 y 19 en la actividad de análisis de rendimiento y las prácticas 21, 22 y 23 en la actividad de análisis de configuración.

### 6.6 SELECCIÓN DE HERRAMIENTAS DE SOPORTE PARA LAS PRÁCTICAS

En esta actividad se realiza la búsqueda de herramientas de soporte para cubrir algunas de las prácticas que la organización aún no tiene cubiertas o las cubre parcialmente y que se pueden apoyar con una herramienta.

En la revisión de las herramientas se realiza un análisis a las prácticas presentes en la TABLA 13 del ANEXO 2 de la cual se seleccionan las prácticas que no se cubren 2 y 5 y de las que se cubren parcialmente se seleccionan las prácticas 3 y 4 para buscar herramientas que les den soporte y mejoren su rendimiento y aplicación.

De acuerdo a los tipos de práctica, se considera necesaria la selección de herramientas de levantamiento de encuestas para la práctica 2, herramienta para modelado de procesos para la práctica 5 y herramientas de soporte y atención a los clientes para generación de incidentes y solicitudes de servicio para las prácticas 3 y 4, además de una herramienta para seguimiento y gestión del proceso de desarrollo de las actividades.

Las herramientas seleccionadas para estas prácticas son las siguientes:

- Práctica 2 (Aplicar Encuestas):**
- OTRS<sup>3</sup>-Survey<sup>4</sup>

<sup>3</sup> OTRS, Software para Mesa de Ayuda y Gestión de Servicios de TI. <http://www.otrs.com/>

<sup>4</sup> OTRS-Survey, Extensión de la herramienta OTRS para generar y aplicar encuestas a los clientes.

TABLA 7 ADAPTACIÓN ENTRE LAS PRÁCTICAS

ID	Práctica	AD <sup>1</sup>	AP
1	Iniciar plan de mejora de calidad y de proceso de mantenimiento		1
24	Diseñar y establecer el funcionamiento del servicio de asistencia/moderador/soporte al cliente		1
2	Aplicar encuestas		1
3	Usar las observaciones, comentarios y quejas para identificar mejoras		1
14	Identificar las VBFs		1
4	Listar y priorizar las mejoras identificadas		1
5	Definir la descripción del proceso de mantenimiento/servicio		2
17	<b>+Planificar el Mantenimiento preventivo.</b>		2
16	Planear e Integrar al proceso pruebas de disponibilidad.	I	2
26	Planificar periodos de revisión en busca de nuevos métodos o funcionamiento.		2
6	Planificar capacitaciones referentes al proceso de mantenimiento		3
7	Planear los tiempos de capacitación		3
8	Identificar procesos de mantenimiento clave y aplicarles métricas		4
9	Establecer objetivos de rendimiento y calidad		4
	<b>+Realizar Análisis de Rendimiento</b>		
13	Realizar análisis de ausencia de disponibilidad	S	4
15	Realizar análisis de SPoF	S	4
19	Prueba de fallas - gestión de fallas	S	4
	<b>+Realizar Análisis de Configuración</b>		
21	Configurar jaulas "chroot" en plataformas Unix	S	4
22	Configurar el mínimo de permisos en sistema de ficheros en todas las plataformas	S	4
23	Si es posible, usar sistemas de fichero de solo lectura	S	4
25	Parchear rápidamente los sistemas.		4
27	Realizar actualizaciones de forma segura		4
10	Identificar e investigar Novedades y mejoras potenciales en el uso del proceso de mantenimiento.	C	5
18	Revisiones y mejoras continuas.	C	5
10	<b>+ Establecer revisiones e identificar e investigar novedades y mejoras potenciales en el uso de proceso de mantenimiento o producto.</b>		
20	Usar los logs (con copias de solo lectura) para identificar problemas y buscar mejoras.	I	5
11	Integrar los nuevos hallazgos* de las investigaciones en el nivel requerido.		5
12	Las mejoras a los procesos actuales, son iniciados de forma contenida localmente		5

<sup>1</sup> AD, Adaptación realizada a la práctica. (I) Integración, (S) Sub actividad y (C) Combinación.

- Google Forms<sup>5</sup>

**Práctica 5** (Definir la descripción del proceso de mantenimiento / servicio):

- Bizagi Process Modeler<sup>6</sup>

**Práctica 3** (Usar las observaciones, comentarios y quejas para identificar mejoras)

<sup>5</sup> Google Forms, Herramienta de google que permite enviar encuestas y almacenar la información en hojas de cálculo.

<sup>6</sup> Bizagi Process Modeler, Aplicación Freeware para diagramar procesos y generar documentación de procesos utilizando la notación estándar BPMN.

- *OTRS Help-Desk*<sup>7</sup>
- *My Agile Board*<sup>8</sup>

#### Práctica 4 (Listar y priorizar las mejoras identificadas)

- *OTRS Help-Desk*
- *My Agile Board*

La selección de las herramientas se hace en base al conocimiento, facilidad de uso y actual uso de las herramientas por parte de la organización.

-La herramienta de *Google Forms* para encuestas fue seleccionada debido a que la organización ya hace uso de encuestas por este tipo de medios y por su facilidad de uso.

-La herramienta *OTRS-Help desk* y *OTRS-Survey*, fueron seleccionadas por medio de un análisis realizado entre diferentes herramientas del mismo tipo y porque presenta el soporte para “*Help desk*” y “*survey*” dentro de la misma herramienta. El benchmarking realizado para esta herramienta es presentado en el ANEXO 3.

-La herramienta *Bizagi Process Modeler* fue seleccionada por que es una de las herramientas usadas actualmente por la organización para el modelado de los procesos en los proyectos.

-La herramienta *My Agile Board*, fue seleccionada por que es una herramienta desarrollada por la organización adaptada a su modelo propio de trabajo en la gestión de los proyectos y seguimiento de las actividades.

## 7 PROPUESTA DE IMPLEMENTACIÓN DEL PROCESO ÁGIL DE GESTIÓN DE MANTENIMIENTO PARA ENTORNOS CLOUD ENFOCADO EN DISPONIBILIDAD

El objetivo de este estudio es definir un proceso de gestión de mantenimiento para ser implementado de manera ágil y ajustado al proceso actual de mantenimiento y desarrollo de software en el área de CC para garantizar la disponibilidad de los productos y de los servicios que se ofrecen.

La presente propuesta de implementación está basada en las mejores prácticas de los modelos S3M, ITIL y OWASP que fueron resultado del análisis de filtros aplicados en este estudio. De las cuales, para S3M se consideraron los niveles de madures 0, 1 y 2, que son 3 de los 6 niveles que lo conforman. Para ITIL, se basa en la gestión de la disponibilidad dentro del área del Diseño del servicio de ITIL v3, y las prácticas del proyecto de seguridad web OWASP.

En esta sección se presenta la propuesta para integrar las prácticas, sus enfoques y las herramientas obtenidas

del estudio realizado dentro de un proceso de mantenimiento ágil de acuerdo a los procesos actuales de la organización. La primera actividad que se realizó fue el análisis de la organización para obtener los procesos actuales como lo propone la metodología MIGME-RRC para identificar sus prácticas internas posteriormente se hace la propuesta de la integración de las prácticas al proceso y después se analiza la cobertura de la propuesta en el proceso actual.

### 7.1 ANÁLISIS DE LA ORGANIZACIÓN

Este análisis se llevó a cabo para obtener el proceso actual de desarrollo y de mantenimiento y soporte al cliente. Este análisis, se llevó a cabo de acuerdo a la propuesta de implementación de MIGME-RRC para *identificar las mejores prácticas internas* de la organización; la cual se enfoca en (1) la extracción del conocimiento tácito de los miembros de la organización y (2) El análisis de la documentación formal de los procesos de la organización [21]. el proceso actual de desarrollo fue extraído de la documentación del proceso de la organización en el proyecto [31], y el proceso de mantenimiento y soporte al cliente fue extraído en base a entrevistas aplicadas al personal de la organización.

#### 7.1.1 PROCESO ACTUAL DE DESARROLLO

El proceso actual de desarrollo de softlogik está basado en *Scrum*<sup>9</sup> y *Extreme Programming*<sup>10</sup>, dos metodologías ágiles para la gestión de los proyectos y un sistema *Kanban* para administrar el proceso de desarrollo de software, en la **Fig. 3** Se muestra el pizarrón Kanban con el proceso actual de desarrollo de software de softlogik, en este pizarrón se muestran en cada columna las diferentes actividades del proceso de desarrollo que debe seguir una pieza de trabajo, la cual es visualizada como una nota o ticket que está redactado en forma de historia de usuario, la cantidad de tickets que pueden estar en una columna está limitado por su trabajo en progreso WIP (por sus siglas en inglés “Work in progress”). En algunas columnas se cuenta con un “check list” o lista de verificación de actividades que debe cumplirse para que las historias de usuario puedan pasar a la siguiente columna. Las actividades que integran el proceso de desarrollo actual de softlogik son las siguientes [31]:

1. **Backlog**, piezas de trabajo vistas como historias de usuario, son todas las funcionalidades que el cliente o dueño del producto desea en el producto final. Esta columna tiene un WIP ilimitado.
2. **Selected**, contiene las historias de usuario que se desarrollaran durante el Sprint.
3. **Development**, contiene las historias de usuario seleccionadas por el equipo de desarrollo para ser desarrollada.

<sup>7</sup> *OTRS-HelpDesk*, herramienta para contacto central entre clientes y empleados para generar incidentes y solicitudes de servicio por medio de tickets. <http://www.otrs.com/es/software/otrs-help-desk/>

<sup>8</sup> *My Agile Board*, Herramienta para gestión y seguimiento de proyectos desarrollada por Softlogik.

<sup>9</sup> Scrum, proceso usado para el desarrollo ágil de software, <http://www.mountaingoatsoftware.com/topics/scrum>.

<sup>10</sup> Extreme Programming, disciplina de desarrollo de software basada en valores de simplicidad, comunicación, retroalimentación y esfuerzo. <http://xprogramming.com/xpmag/whatixp>.

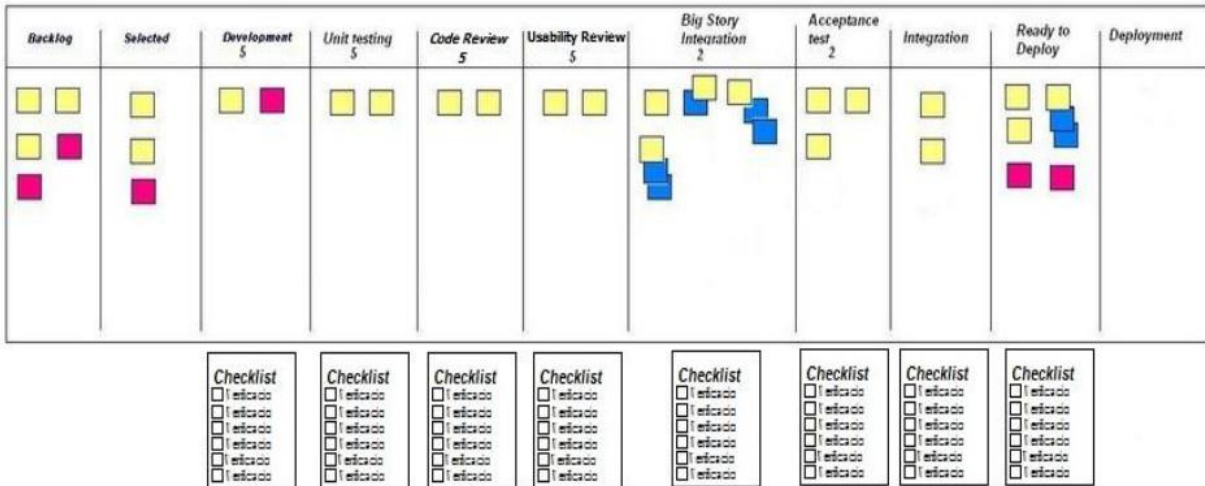


Fig. 3 Última versión del proceso de desarrollo de Softlogik.

Fuente: [31]

4. **Unit testing**, se realizan las pruebas siguiendo la metodología de desarrollo guiado por pruebas o *Test Driven Development*. Se prueban requerimientos funcionales o procedimientos especiales.
5. **Code Review**, revisión del código fuente por un miembro del equipo que no estuvo involucrado en el desarrollo de esa historia de usuario.
6. **Usability Review**, una persona externa al equipo de desarrollo evalúa la usabilidad de la funcionalidad desarrollada.
7. **Big Stroy Integration**, las historias de usuario que tienen dependencias o fueron divididas en sub-tareas son apiladas en espera de que se complete cada una de ellas para ser integradas en la historia de usuario original.
8. **Acceptante test**, se prueba la funcionalidad del producto, estas pruebas son ejecutadas automáticamente.
9. **Integration**, ejecución de pruebas programadas para garantizar la integridad del producto de manera incremental asegurándose de que cada nueva funcionalidad no rompe al producto.
10. **Ready to Deploy**, se realizan pruebas en un servidor de desarrollo con las mismas características que un servidor de producción.
11. **Deployment**, cuando el nuevo reléase has sido probado con éxito en el servidor de desarrollo se pasa al servidor de producción.

### 7.1.2 PROCESO ACTUAL DE MANTENIMIENTO

En esta actividad se describe el proceso actual de mantenimiento que usa la organización, este proceso fue extraído en base a entrevistas realizadas al Director de operaciones y a un Ingeniero Sénior de la organización.

El proceso actual de mantenimiento de la organiza-

ción está basado en un proceso de atención y servicio de soporte al cliente y un conjunto de actividades que son realizadas de acuerdo a las necesidades de cada proyecto y sin ningún seguimiento estandarizado. Algunas de estas actividades relacionadas a mantenimiento en el área de disponibilidad ya fueron analizadas en el punto 6.2 Validación de las Prácticas Resultantes, donde se identifican las prácticas que ya son llevadas a cabo por la organización.

El proceso actual de mantenimiento para atención y soporte al cliente es modelado en la Fig. 4. Debido a que softlogik, no cuenta con un departamento dedicado al mantenimiento y tiene que tener a su personal asignado a los proyectos actuales, la organización definió un proceso de atención, para llevar a cabo este proceso de atención, los líderes de equipo asignan a un responsable de mantenimiento por proyecto, este responsable es el ingeniero que haya tenido una mayor participación en el desarrollo del proyecto asignado, sin embargo, si no existen peticiones de mantenimiento para ese proyecto, el ingeniero asignado debe estar trabajando en su proyecto actual.

Cuando un cliente envía una petición de mantenimiento que puede ser: (1) un incidente, (2) un cambio, o un (3) un problema, es primeramente revisado por el ingeniero de mantenimiento asignado como responsable a ese proyecto, quien puede dejar su proyecto actual para atender la petición de mantenimiento, si el cliente que hace la petición no es la persona autorizada para solicitar ese tipo de peticiones, se solicita la autorización a la persona responsable por parte del cliente, en caso de no requerir validaciones adicionales, el ingeniero verifica si puede solventar esa petición, en caso de que no pueda, se lo notifica al líder de proyectos, y es este quien asigna un nuevo responsable, donde este nuevo responsable, es la persona que participó específicamente en el desarrollo del módulo, componente o servicio requerido por la petición.



**TABLA 8 RUTA DE SEGUIMIENTO DE LAS PRÁCTICAS**

AP	Práctica	Tipo <sup>1</sup>	Backlog	Selected	Development	Unit Testing	Code Reviewer	Usability Review	Big Story Integration	Acceptance test	Integration	Ready to deploy	Deployment
AP1	1	1,2,3,4	o	o	o			o		o		o	o
AP1	24	1,2,3	o	o	o			o		o		o	o
AP1	2	1,2,3	o	o	o	o		o		o		o	o
AP1	3	1,2,3	o	o	o	o		o		o		o	o
AP1	14	1,2,3	o	o	o			o		o		o	o
AP1	4	1,2,3	o	o	o			o		o		o	o
AP2	5	1,2,3	o	o	o	o		o	o	o	o	o	o
AP2	17	1,2,3	o	o	o	o		o		o		o	o
AP2	26	1,2,3	o	o	o			o		o		o	o
AP3	6	1,2,3	o	o	o			o		o		o	o
AP3	7	1,2,3	o	o	o			o		o		o	o
AP4	8	1,2,3	o	o	o			o		o		o	o
AP4	9	1,2,3,4	o	o	o	o		o		o		o	o
AP4	13,15,19	1,3	o	o	o	o		o		o		o	o
AP4	21,22,23	1,2,3	o	o	o	o		o		o		o	o
AP4	25	2	o	o	o	o		o	o	o	o	o	o
AP4	27	1,2	o	o	o	o		o	o	o	o	o	o
AP5	10	1,2,3	o	o	o	o		o		o		o	o
AP5	18	1,2,3	o	o	o	o		o		o		o	o
AP5	11	1,2,3	o	o	o	o		o		o	o	o	o
AP5	12	1,2,3	o	o	o	o		o		o	o	o	o

<sup>1</sup> Tipo de plataforma de Cloud. (1)Infraestructura, (2) Plataforma, (3) Software, (4) Implica realizar la tarea involucrando el nivel directivo.

que se complete cada una de ellas para ser integradas en la práctica original.

7. **Acceptante test**, Se realizan las pruebas de aceptación, estas pruebas son realizadas por el equipo de desarrollo y en caso de ser posible, se programan automáticamente.
8. **Ready to Deploy**, Se realizan pruebas en un servidor de desarrollo con las mismas características que un servidor de producción, en caso de referirse a procesos, estos se validan con la organización en pruebas de forma contenida para verificar su aceptación y adaptabilidad.
9. **Deployment**, Cuando el nuevo reléase has sido probado con éxito en el servidor de desarrollo se pasa al servidor de producción o el área destinada a implementar la mejora.

### 7.3 COBERTURA DE LA PROPUESTA EN EL PROCESO ACTUAL

En esta actividad se realiza el análisis de como las prácticas son cubiertas por las actividades y procesos actuales de la organización y como las prácticas parciales pueden ser cubiertas.

De acuerdo a los resultados de validación de las prácticas resultantes de la aplicación de los filtros, con respecto a las prácticas finales del proceso de integración y mejora, se obtuvieron un total de 26 prácticas finales, las cuales son descritas y modeladas en el ANEXO 4. Con respecto a la cobertura de las prácticas con el proceso actual de la organización, se encontró que la empresa cubre completamente 10 prácticas (38.46%), 7 son cubiertas de forma parcial (26.92%) y 9 prácticas (34.62%) no son considera-

**TABLA 9** PROPUESTA DE COBERTURA DE PRÁCTICAS NO CUBIERTAS O PARCIALES

ID	Propuesta de cómo pueden ser resueltas las prácticas
1	Mediante la aplicación de un programa de mantenimiento específico en cubrir el enfoque designado a esta práctica.
24	Mediante el diseño y establecimiento de un plan de atención al cliente que permita identificar los tipos de problemas atendidos de acuerdo a los problemas relacionados con ayuda de la herramienta OTRS-HelpDesk
2	Mediante la aplicación de encuestas en las que se incluirán preguntas orientadas a cubrir puntos específicos de los problemas relacionados con apoyo de herramientas de encuestas como Google Forms y OTRS-Survey
3	Mediante la implementación de la herramienta OTRS-HelpDesk, un repositorio de issues (tickets) que permita el registro, seguimiento y la correcta identificación con respecto a los problemas, las observaciones, comentarios y quejas.
4	Mediante la categorización de las incidencias recogidas en la herramienta OTRS-HelpDesk, las categorías permitirán identificar los problemas más comunes y recurrentes lo que permitirá definir planes de acción y su prioridad. La categorización se hará de acuerdo a los 3 principales problemas relacionados con apoyo de la herramienta OTRS-HelpDesk
5	Mediante la definición del proceso de mantenimiento, tomando como base el proceso actual y el resultado de aplicar las prácticas resultantes de este estudio.
17	Mediante el aseguramiento de que se definan actividades de mantenimiento preventivas enfocadas en cubrir cada uno de los problemas relacionados.
16	Si no se presentan en los requisitos de los proyectos, se solventaría mediante la planeación e integración de pruebas que cubran cada uno de los problemas relacionados.
26	Mediante el establecimiento de un plan de pruebas periódicas que permita identificar problemas en la disponibilidad del servicio, este plan será enfocado en cubrir los problemas relacionados.
6	Mediante la implementación de un plan de capacitación sobre estrategias de aseguramiento de la disponibilidad del servicio, y que permitan cubrir los 3 problemas.
8	Mediante la aplicación métricas a las actividades principales del proceso de mantenimiento definido para la organización.
9	Mediante el establecimiento de los objetivos de rendimiento y calidad esperados para el servicio de mantenimiento.
21	Mediante la definición de un plan de configuración cuando se utilicen plataformas Unix y se considere necesario de acuerdo a los requerimientos.
23	Mediante la definición de un plan de configuración de ficheros de solo lectura y se considere necesario de acuerdo a los requerimientos.
10	Enfocando algunas de las actividades que ya se hacen en cada uno de los problemas relacionados.
20	Mediante la definición de un plan de consulta que asegure que las consultas se hacen sin posibilidades de alterar los logs.

das por la organización, estos resultados se muestran en la gráfica de la Fig. 6.

En la TABLA 9, se muestra el resumen de algunas prácticas que no son cubiertas completamente y la propuestas de cómo podrían ser cubiertas con el proceso actual, aplicando el enfoque seleccionado y sus herramientas.

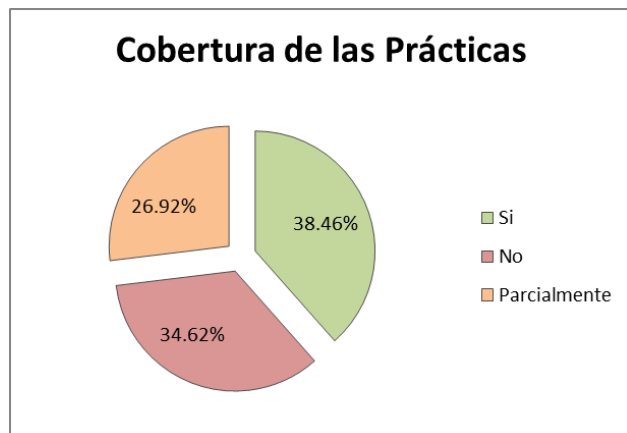


Fig. 6 .Gráfica de cobertura de las prácticas.

## 8 CONCLUSIONES Y TRABAJO FUTURO

Para la presente investigación se realizó una revisión de la literatura actual, estudiando y analizando trabajos

relacionados con procesos, prácticas de mantenimiento de software, técnicas y modelos para aseguramiento de la disponibilidad de la información en "Cloud Computing".

Como resultado de la investigación, se elaboró una propuesta como solución inicial, la cual propone integrar un proceso de gestión de mantenimiento basado en prácticas de los modelos S3M, ITIL y OWASP resultantes de aplicar los filtros de mantenimiento, disponibilidad y manifiesto ágil para asegurar el atributo de calidad de disponibilidad en los servicios de Cloud Computing, la implementación de este proceso se basa en los procesos actuales de desarrollo y de mantenimiento de la organización.

Con el análisis realizado a esta propuesta y a las prácticas resultantes, se identificó un nivel de madurez de la organización, ya que de acuerdo a la cobertura de las prácticas finales se obtuvo una cobertura del 38.4% en prácticas totalmente cubiertas, 26.9% de prácticas que se cumplen parcialmente y solo un 34.6% de prácticas no se realizan por la organización. Por lo que para la organización solo quedaría aplicar mayor esfuerzo en las prácticas que no se cubren ya que las prácticas que se cubren parcialmente pueden ser fácilmente cubiertas aplicando el enfoque necesario a satisfacer los problemas relacionados a la disponibilidad de la información.



Las principales limitantes de este proyecto con respecto al cumplimiento del objetivo, es que al ser una propuesta inicial, aún es necesario validar este proceso con proyectos reales en la organización para poder determinar su efectividad.

Como trabajo futuro, se pretende realizar la validación del proceso actual con proyectos reales en la organización y completar la investigación de las prácticas de los modelos S3M e ITIL en todos sus niveles, así como complementar con otros modelos y normas afines a procesos de mantenimiento en el área de infraestructura y plataforma de Tecnologías de la Información para lograr una mayor cobertura de las capas de abstracción que comprende el "Cloud Computing".

Considerando que en esta propuesta inicial se propone que todas las prácticas se conviertan en tareas y sigan una ruta específica dentro del Kanban, se considera que es posible que no todas las prácticas se deban convertir a tareas, por lo que en trabajos futuros, se pretende experimentar integrando algunas prácticas a los checklists de las etapas e inclusive, considerar la posibilidad de que también se pudieran manejar completamente como una etapa individual dentro del Kanban.

Otra consideración importante en trabajos futuros es la formalización del proceso de filtrado de las prácticas, ya que el trabajo realizado en este estudio para filtrar las prácticas estuvo basado en una selección conceptual y de cobertura de acuerdo a la relación de la práctica con la definición del criterio de filtrado, por lo que se considera importante complementar y formalizar en trabajos posteriores este proceso de filtrado.

## AGRADECIMIENTO

Le agradezco a Dios por haberme permitido concluir esta etapa importante de mi vida, a mis Familiares, amigos y compañeros que siempre me apoyaron durante este proceso, a toda la comunidad CIMAT Zacatecas y en especial a mis sinodales La Dra. Alejandra, Pepe Hernández y Jezreel Mejía por su apoyo y revisiones a este trabajo, Al Dr. Hugo A. Mitre por su apoyo y participación en la elaboración de este documento y a Pina García por su apoyo incondicional y por darse el tiempo de hacer revisiones al contenido de este documento.

## REFERENCIAS

- [1] L. M. Vaquero, L. Rodero-merino, J. Caceres, and M. Lindner, "A Break in the Clouds : Towards a Cloud Definition," *Computer Communication Review*, vol. 39, no. 1, pp. 50-55, 2009.
- [2] S. Zardari, "Cloud Adoption : A Goal-Oriented Requirements Engineering Approach," *Requirements Engineering*, pp. 29-35, 2011.
- [3] W. Y. C. Wang, A. Rashid, and H.-M. Chuang, "TOWARD THE TREND OF CLOUD COMPUTING," *Journal of Electronic Commerce Research*, vol. 12, no. 4, pp. 238-243, 2011.
- [4] M. Kaiserswerth, O. Brian, T. Brunschwiler, H. Dill, H. Christ, B. Falsafi, M. Fischer, S. Gatzui Grivas, and C. Giovanoli, "SATW Cloud computing," 2007). [http://download.boulder.ibm.... Zürich](http://download.boulder.ibm....Zürich), p. 51, 2007.
- [5] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *Ieee Transactions On Dependable And Secure Computing*, vol. 1, no. 1, pp. 11-33, 2004.
- [6] D.-W. Sun, G.-R. Chang, S. Gao, L.-Z. Jin, and X.-W. Wang, "Modeling a Dynamic Data Replication Strategy to Increase System Availability in Cloud Computing Environments," *Journal of Computer Science and Technology*, vol. 27, no. 2, pp. 256-272, Mar. 2012.
- [7] A. Bala and I. Chana, "Fault Tolerance- Challenges , Techniques and Implementation in Cloud Computing," *International Journal of Computer Science*, vol. 9, no. 1, pp. 288-293, 2012.
- [8] Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid, "Availability and Load Balancing in Cloud Computing," vol. 14, pp. 134-140, 2011.
- [9] M. Armbrust, A. D. Joseph, R. H. Katz, and D. A. Patterson, "Above the clouds: A Berkeley view of cloud computing," ... , *Berkeley, Rep. ....*, 2009.
- [10] P. Grubb and A. Takang, *Software maintenance: concepts and practice*. 2003.
- [11] R. Hasan, S. Chakraborty, and J. Dehlinger, "Examining Software Maintenance Processes in Small Organizations : Findings from a Case Study," pp. 129-143, 2012.
- [12] S. Engineering and S. Committee, *IEEE Standard for Software Maintenance*, vol. 1998. 1998.
- [13] S. Taylor, V. Lloyd, and C. Rudd, "ITIL Version 3 Service Design," *The Office of Government Commerce*, 2007.
- [14] T. Chitra, "Life based maintenance policy for minimum cost," *Annual Reliability and Maintainability Symposium*, 2003., pp. 470-474, 2003.

- [15] B. Hunt, B. Turner, and K. Mcritchie, "Software Maintenance Implications on Cost and Schedule," vol. 20, no. 12, pp. 1-6, 2008.
- [16] S3M, "Process Management ( PRO ) BEST PRACTICES FROM LEVEL 0,1 and 2," pp. 1-34, 2008.
- [17] M. Mirna and M. Jezreel, "The results analysis of using MIGME-RRC methodology for software process improvement," ... (CISTI), 2011 6th ..., no. 852, 2011.
- [18] A. C.-M. Jose, C. Gonzalo, M. Mirna, and S. F. Tomas, "Metodología para la Implementación Gradual de Mejoras, con Enfoque en la Reducción de la Resistencia al Cambio," *Proceedings 4ª Conferencia Ibérica de Sistemas y Tecnologías de la Información (CISTI 2009)*, 2009.
- [19] M. Kajko-mattsson and J. Nyfjord, "A Model of Agile Evolution and Maintenance Process," pp. 1-10, 2009.
- [20] A. Wiesmann, A. van der Stock, M. Curphey, and R. Stirbei, "A guide to building secure web applications and web services," *The Open Web Application Security Project (OWASP)*, 2005.
- [21] J. A. Calvo Manzano, G. Cuevas, G. Gómez, J. Mejía, M. Muñoz, and S. F. Tomás, "Methodology for process improvement through basic components and focusing on the resistance to change," *Journal of Software Maintenance and Evolution: Research and Practice*, 2010.
- [22] M. Seikola, H.-M. Loisa, and A. Jagos, "Kanban Implementation in a Telecom Product Maintenance," *2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications*, pp. 321-329, Aug. 2011.
- [23] P. Middleton and D. Joyce, "Lean Software Management: BBC Worldwide Case Study," *IEEE Transactions on Engineering Management*, vol. 59, no. 1, pp. 20-32, Feb. 2012.
- [24] D. Anderson and D. Dumitriu, "From Worst to Best in 9 Months," *TOC ICO World Conference*, no. November, 2005.
- [25] L. Gan, X. Deng, Q. Tian, and R. Wang, "Explores of maintenance system based on Cloud Computing," *2012 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pp. 530-532, Jun. 2012.
- [26] B. Radojevic and M. Zagar, "Analysis of issues with load balancing algorithms in hosted (cloud) environments," *MIPRO, 2011 Proceedings of the 34th ...*, pp. 416-420, 2011.
- [27] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google file system," *ACM SIGOPS Operating Systems Review*, vol. 37, no. 5, p. 29, 2003.
- [28] The Open Web Application Security Project (OWASP), "Una Guía para Construir Aplicaciones y Servicios Web Seguros," *2001 - 2005 Free Software Foundation.*, 2005.
- [29] M. Barbacci, M. Klein, T. Longstaff, and C. Weinstock, "Quality Attributes.," *CMU/SEI-95-TR-021*, 1995.
- [30] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, "Principios del Manifiesto Ágil." .
- [31] CIMAT, "PROYECTO: Red de Innovación para la Planeación, Ejecución, Control y Mejora Continua de Procesos Orientados al Desarrollo de Software en el Mercado Regional Basado en Métodos Ágiles," unpublished.

## ANEXOS

### ANEXO 1 RESULTADOS DE APLICACIÓN DE LOS FILTROS

**TABLA 10** DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A S3M

Modelo	Área	#	Práctica	M <sup>1</sup>	D <sup>2</sup>	MA <sup>3</sup>	PAR <sup>4</sup>	S <sup>5</sup>	Tipo <sup>6</sup>
S3M	PRO1	1	Un programa de mejora de la calidad y del proceso ha sido iniciado en el departamento de TI a nivel organizacional. Los gerentes del área de mantenimiento de software están enterados de este programa y han recibido un entrenamiento de concientización inicial sobre la calidad.	Ok	Ok	Ok	5	Ok	1,2,3,4
		2	Los resultados de encuestas de mantenimiento de software aplicados a los clientes sobre los productos/servicios son usados para identificar mejoras.	Ok	Ok	Ok	4,12	Ok	1,2,3
		3	Las observaciones, comentarios y quejas de los usuarios/clientes y los grupos de interfaces son recolectados y usados para identificar mejoras.	Ok	Ok	Ok	4,12	Ok	1,2,3
		4	Los datos de las fallas del software son recolectados y usados para identificar mejoras al proceso de mantenimiento/producto y también a las interfaces con los otros grupos de interfaces.	Ok	Ok	x		x	
		5	Se realiza una lista con las mejoras candidatas a implementarse ordenadas por prioridades, y son usadas como guía para desarrollar el programa de mejora del mantenimiento de software.	Ok	Ok	Ok	2,7,9,12	Ok	1,2,3
		6	Las 10 posibles mejoras de la lista son discutidas, se realizan planes de acción por los medios y altos directivos. Las actividades/proyectos planeadas son soportadas dentro del presupuesto anual de mantenimiento y operaciones actuales.	Ok	Ok	x		x	
	PRO2	7	Exista al menos una descripción del proceso de mantenimiento/servicio que es usado por el cliente y es ofrecido por la organización.	Ok	Ok	Ok	9	Ok	1,2,3
		8	Partes del proceso de mantenimiento/servicios/recursos son documentados e implementados en la organización.	Ok	Ok	x		x	
	PRO3	9	El ingeniero de mantenimiento de software es entrenado y motivado a realizar bien las cosas cuando usa el proceso/servicio y ejecuta su rol.	Ok	Ok	Ok	5	Ok	1,2,3
		10	Existen planes y registros que describen el nivel educativo y entrenamiento necesario para cada puesto de ingeniero de mantenimiento y de aplicaciones de software. Este plan de entrenamiento incluye capacitaciones necesarias, cursos ofrecidos, y material relacionado, créditos, recursos disponibles y el plan educativo y actividades de entrenamiento.	Ok	Ok	x		x	
		11	El tiempo de capacitación es planeado.	Ok	Ok	Ok	9,12	Ok	1,2,3
	PRO4	12	Algunos procesos de mantenimiento clave, servicios y productos son medidos. Estos son definidos y usados localmente.	Ok	Ok	Ok	12	Ok	1,2,3
		13	Líneas base de medida de la calidad y rendimiento del proceso, servicios, y productos son recolectados, almacenados, revisados y usados con los stakeholders. Estos datos son usados para mejorar el proceso actual, los servicios y/o productos.	Ok	Ok	x		x	
		14	La organización de mantenimiento ha establecido algunos objetivos de rendimiento y calidad.	Ok	Ok	Ok	5,9	Ok	1,2,3,4
	PRO5	15	Nuevos procesos, servicios, tecnologías, metodologías y herramientas son identificados e investigados por su potencial uso en el mantenimiento de software.	Ok	Ok	Ok	12	Ok	1,2,3
		16	Nuevos procesos, servicios, tecnologías, metodologías y herramientas son alcanzados e introducidos en el nivel requerido.	Ok	Ok	Ok	12	Ok	1,2,3
		17	Mejoras a algunos procesos de mantenimiento, servicios, tecnologías, metodologías y herramientas han sido iniciados de una forma controlada localmente.	Ok	Ok	Ok	12	Ok	1,2,3

<sup>1</sup> Mantenimiento

<sup>2</sup> Disponibilidad

<sup>3</sup> Manifiesto Ágil

<sup>4</sup> Práctica Ágil Relacionada

<sup>5</sup> Seleccionada finalista

<sup>6</sup> Tipo de plataforma de Cloud.

1) IaaS

2) PaaS

3) SaaS

4) Implica realizar la tarea involucrando el nivel directivo.

TABLA 11 DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A ITIL

Modelo	Área	#	Práctica	M	D	MA	PAR	S	Tipo
ITIL-v3 Service Design Gestión de la Disponibilidad	Actividades Reactivas	1	Monitorear, medir, analizar y reportar la disponibilidad de los componentes y el servicio.	Ok	Ok	X		X	
		2	Análisis de ausencia de disponibilidad.	Ok	Ok	ok	12	ok	1,3
		3	Ciclo de vida de incidentes expandido (The expanded incident lifecycle).	X	x	x		x	
		4	Análisis de fallas en el Servicio.	Ok	Ok	X		X	
	Actividades Proactivas	5	Identificar las funciones vitales del negocio o VBFs (Vital Business Functions).	Ok	Ok	ok	9	ok	1,2,3
		6	Diseñar para disponibilidad.	Ok	Ok	X		X	
		7	Análisis de impacto de fallas en los componentes.	Ok	Ok	X		X	
		8	Análisis de punto único de fallo o SPoF (Single Point of Failure).	Ok	Ok	ok	9	ok	1,3
		9	Análisis de fallas en árbol o TFA (Fault Tree Analysis).	X	x	x		x	
		10	Modelado.	Ok	Ok	X		X	
		11	Gestión y análisis de Riesgos	Ok	Ok	X		X	
		12	Programación de pruebas de disponibilidad.	Ok	Ok	ok	9,12	ok	1,2,3
		13	Mantenimiento planificado y preventivo.	Ok	Ok	Ok	9,12	ok	1,2,3
		14	Elaboración del documento de proyección de interrupción en los servicios o PSO (Projected Service Outage).	X	x	x		x	
		15	Revisiones y mejoras continuas.	Ok	Ok	ok	12	ok	1,2,3

TABLA 12 DETALLE DE RESULTADOS DE APLICACIÓN DE FILTROS A OWASP

Modelo	Área	#	Práctica	M	D	MA	PAR	S	Tipo
OWASP - Development guide	Autenticación	1	La autenticación es solo tan fuerte como sus procesos de administración de usuarios	X	x	x		x	
		2	Use la forma más apropiada de autenticación adecuada para su clasificación de bienes.	X	x	x		x	
		3	Re-autenticar al usuario para transacciones de alto valor y acceso a áreas protegidas	X	x	x		x	
		4	Autenticar la transacción, no el usuario	X	x	x		x	
		5	Las contraseñas son trivialmente rotas y no son adecuadas para sistemas de alto valor.	X	x	x		x	
	Manejo de Sesiones	6	Considerar cuidadosamente donde guarda el estado de la aplicación	ok	ok	x		x	
		7	Los datos contenidos en formularios de varias páginas pueden ser enviados de vuelta al usuario en los siguientes dos casos: 1. Cuando existen controles de integridad para prevenir la manipulación. 2. Cuando los datos son validados luego de cada envío del formulario, o al menos al final del proceso de envío.	x	x	x		x	
	Manejo de errores, auditoría y Logs	8	Los secretos de la aplicación (tales como credenciales del lado del servidor e información sobre roles) nunca debería ser visible al cliente. Estos deben ser guardados en una sesión o del lado del servidor.	x	x	x		x	
		9	Prueba de fallas – no deje sin gestionar las fallas.	ok	ok	ok	12	ok	1,3
		10	Utilice logs de doble propósito	ok	ok	x		x	
		11	Los logs de auditoría se encuentran legalmente protegidos – protéjalos!	ok	x	x		x	
		12	Genere reportes y realice búsquedas en los logs utilizando una copia de solo lectura o una copia íntegra del original.	ok	ok	ok	9,12	ok	1,2,3

Modelo	Área	#	Práctica	M	D	MA	PAR	S	Tipo
Sistema de Archivos		13	Usar jaulas “chroot” en plataformas Unix.	ok	ok	ok	9	ok	1,2,3
		14	Usar los mínimos permisos en el sistema de ficheros de todas las plataformas.	ok	ok	ok	9	ok	1,2,3
		15	Considerar la utilización de sistemas de fichero de solo lectura (como CD-ROM o llaves USB bloqueadas) si fuera posible.	ok	ok	ok	9	ok	1,2,3
Interfaces Administrativas		16	Cuando se está diseñando aplicaciones, trazar la funcionalidad administrativa fuera y asegurarse que los controles apropiados de acceso y auditoria están en su lugar	x	x	x		x	
		17	Considerar procesos – en algunas ocasiones todo lo que se requiere es entender como los usuarios pueden ser prevenidos de utilizar una característica con la simple falta de acceso.	x	x	x		x	
		18	Acceso de servicio de asistencia es siempre un término medio – ellos necesitan acceso para ayudar a los clientes, pero no son administradores.	x	x	x		x	
		19	Diseñar cuidadosamente la funcionalidad de servicio de asistencia / moderador / soporte al cliente alrededor de una capacidad administrativa limitada y aplicación segregada o acceso.	ok	ok	ok	9,4	ok	1,2,3
Configuración		20	Desactivar todas las opciones innecesarias de manera predeterminada	x	x	x		x	
		21	Asegurar que todas las opciones y configuraciones para cada función están inicialmente configuradas para ser la elección más segura posible.	x	x	x		x	
		22	Inspeccionar el diseño para comprobar si las elecciones menos seguras pudieran ser diseñadas de otra manera. Por ejemplo, los sistemas de restablecimiento de contraseña son pobres desde el punto de vista de seguridad. Si no proporciona este componente, los usuarios de su aplicación estarán más seguros.	x	x	x		x	
		23	No confíe en características instaladas opcionalmente en el código base.	x	x	x		x	
		24	No configure nada como preparación para una característica opcional de implantación.	x	x	x		x	
Mantenimiento		25	Es responsabilidad como desarrollador garantizar que el usuario este tan seguro como sea posible y animar a parchear rápidamente los sistemas vulnerables para garantizar que sus parches sean integrales (es decir, no más parches de este tipo), regresión de problemas anteriores (es decir, soluciones permanecen solventadas), y estable (es decir, que se han realizado suficientes ensayos).	ok	ok	ok	1,3	ok	2
		26	Las aplicaciones deben mantenerse con regularidad, en busca de nuevos métodos para evitar los controles de seguridad	ok	ok	ok	9,12	ok	1,2,3
		27	Es normal dentro de la industria proporcionar apoyo desde n-1 hasta n-2 versiones, por lo que se requiere algún tipo revisión de control de código fuente, como CVS, ClearCase, o subversión, son necesarios para gestionar las correcciones de seguridad y evitar errores de regresión.	ok	x	x		x	
		28	Las actualizaciones deben ser prestadas de forma segura, ya sea por firma digital de los paquetes, o mediante un resumen de mensaje libre de colisiones.	ok	ok	ok	9	ok	1,2
		29	El apoyo a la política de parches de seguridad deben ser claramente comunicada a los usuarios, para asegurar que los usuarios conocen que versiones están soportadas para soluciones de seguridad y cuando termina su vida.	ok	x	x		x	

## ANEXO 2 RESULTADOS DE LA VALIDACIÓN DE LAS PRÁCTICAS EN LA ORGANIZACIÓN

TABLA 13 RESULTADOS DE LA VALIDACIÓN DE LAS PRÁCTICAS EN LA ORGANIZACIÓN

ID	Problema	¿Es implementada?	¿Desde cuándo?	¿Existe algún beneficio?	Como se implementa Actualmente
1	1,2,3	No	-	-	El personal capacitado se encarga de la configuración de los equipos e infraestructura necesaria para tener el servicio disponible. Además de que por el momento no se tienen planes de investigaciones sobre disponibilidad de servicios.
2	1,2,3	No	-	-	No se aplican encuestas, el equipo está en constante comunicación con el cliente.
3	1,2,3	<b>*Parcialmente,</b> Se obtienen los comentarios directamente de los clientes pero se manejan de forma general, por lo que no se puede identificar a que problema atiende.	Feb, 2012	Se mejora continuamente y se gana confianza del cliente en nuestro servicio.	Siempre se toma en cuenta la retroalimentación recibida de los usuarios de los servicios/productos. Los desarrolladores trabajan directamente con los clientes y obtienen los comentarios directamente de ellos. En caso de que las observaciones requieran de cambios mayores, deben pasara por el gerente y ser autorizados. Aunque no se genera una lista de mejoras en base a esta retroalimentación recibida.
4	1,2,3	<b>*Parcialmente,</b> Las mejoras no se listan ni se priorizan, por lo que solo puede identificar el problema que atiende.	Feb, 2012	La mejora continua y la confianza de los clientes en nuestro servicio.	No se hacen planes de mejora, los datos son recolectados parcialmente pero no se priorizan, solo se identifican aquellos que sean relevantes o aplicables inmediatamente de acuerdo a la criticidad para el rendimiento del proceso/servicio.
5	1,2,3	No	-	-	-
6	1,2,3	<b>*Parcialmente,</b> Las capacitaciones se programan de acuerdo a los requerimientos y necesidades de los proyectos, no se enfocan en cubrir directamente los problemas relacionados.	Feb, 2012	Se ofrecen mejores servicios de mantenimiento y soporte a los clientes.	Constantemente se llevan a cabo capacitaciones para el equipo sobre los procesos que manejan y diferentes tecnologías.
7	1,2,3	<b>*Completo,</b> Se planean tiempos de capacitación de acuerdo al problema que se esté tratando, en caso que existiera capacitación para algún problema relacionado, también se planearía de igual forma.	Feb, 2012	Reducción en los costos de capacitación por que se enfocan sol al equipo/persona que lo necesita.	De acuerdo al proyecto que se está manejando, se capacita al personal que estará involucrado.
8	1,2,3	No	-	-	-
9	1,2,3	No	-	-	-
10	1,2,3	<b>*Parcialmente,</b> La actividad no se enfoca en ninguno de los problemas relacionados.	Feb, 2012	Mejora de los procesos actuales de la organización y los servicios que se ofrecen a los clientes.	Siempre se está en busca de nuevas tecnologías para implementar proyectos, procesos y/o tecnologías emergentes. Aunque las investigaciones actualmente no están orientadas en el aseguramiento de la calidad de la disponibilidad del servicio.
11	1,2,3	<b>*Completo,</b> En caso que existieran nuevas integraciones para algún problema relacionado, se haría la integración de igual forma.	Feb, 2012	Innovación en el servicio y un equipo mejor capacitado.	Se cuenta con un grupo de especialistas que realizan las pruebas de la implementación de nuevos procesos, servicios, tecnologías, metodologías o herramientas. Estas pruebas son realizadas directamente en el nivel o área al que corresponden y si se obtienen buenos resultados, se procede a integrarlos a producción.
12	1,2,3	<b>*Completo,</b> En caso que existieran mejoras para integrar referentes a algún problema relacionado, se haría la integración de la mejora de igual forma.	Feb, 2012	Un continuo y mejor control del proceso.	Desde que se realiza la evaluación, las mejoras se van monitoreando para integrarlas en el área específica.

ID	Problema	¿Es implementada?	¿Desde cuándo?	¿Existe algún beneficio?	Como se implementa Actualmente
13	2	<b>*Completo,</b> Se realizan varios tipos de pruebas al sistema que permiten determinar la disponibilidad de la aplicación y la respuesta/tolerancia a los fallos.	Ago., 2011	Identificar y definir las necesidades para asegurar que el servicio cubra las expectativas de los clientes.	Se realizan pruebas de seguridad, penetración, disponibilidad, performance/desempeño en la infraestructura que pone el cliente, principalmente para identificar cuantos usuarios va a soportar el sistema. Se genera un dictamen y en base a estos resultados se identifica en que está limitando la disponibilidad del servicio.
14	2	<b>*Completo,</b> La empresa SLK y Compulogic realizan un análisis para identificar los VBFs y las restricciones de recursos que pudieran tener.	Ago., 2011	En la identificación e implementación de un plan de acción para identificar la infraestructura necesaria para el funcionamiento de la aplicación.	Mediante un análisis en conjunto hecho por SLK y Compulogic en base a su experiencia para identificar la infraestructura actual y el uso esperado de esa infraestructura. Se generan las recomendaciones por escrito.
15	1	<b>*Completo,</b> Mediante el análisis realizado por las empresas SLK y softlogik, se identifican los puntos de falla y se establecen planes de acción, en caso de ser necesario, se contrata a terceros el servicio para asegurar la correcta gestión y tolerancia a fallas.	Ago., 2011	Se asegura el nivel de servicio ofrecido al tercerizarse, al ofrecer menos de lo que se contrata se asegura la disponibilidad Ofrecida.	Mediante un análisis en conjunto de SLK y Compulogic, se evalúan las plataformas requeridas para identificar los posibles puntos de fallo, en caso de ser necesario se contratan SLA's para ofrecer servicios de Cloud. En base al servicio contratado (ej. 99.99 de disp.) es el nivel de disponibilidad que se le ofrece al cliente considerando ofrecer menos de lo que se contrata para dejar un colchón de ajuste (99.00).
16	1,2,3	<b>*Parcialmente,</b> Se planea e integran pruebas de disponibilidad pero solo para satisfacer los requisitos específicos del proyecto.	-	Permite identificar posibles puntos de falla y/o limitaciones del servicio ofrecido o usado por los clientes.	En función de los requerimientos no funcionales del proyecto se identifican las necesidades en infraestructura y plataforma que soporten estos requerimientos y se establecen las pruebas conforme a estos requerimientos para hacer las pruebas de disponibilidad.
17	1,2,3	<b>*Parcialmente,</b> Se cubre el mantenimiento preventivo, con el análisis de los requerimientos, planes de acción para prevenir fallas, investigación de nueva tecnología y mejoras, pero no existe enfoque definido en solventar los problemas relacionados.	-	Se ajustan el servicio y planes de disponibilidad de acuerdo a las necesidades del cliente.	Conforme al tipo de proyecto, se establecen los requerimientos no funcionales y se define el tipo de atención necesario para satisfacer esos requerimientos para abordar directamente las necesidades y puntos críticos del proyecto.
18	1,2,3	<b>*Parcialmente,</b> Cumple con la identificación e integración de mejoras y nuevas tecnologías, pero no se tiene el suficiente manejo de seguimiento y monitoreo a los procesos/servicios/tecnologías usados por los clientes.	-	Se identifican los requerimientos de la infraestructura y plataforma de los clientes y se les pueden hacer propuestas de mejoras.	Mediante el establecimiento de requerimientos actuales y requerimientos a futuro para anticiparse a las posibilidades de crecimiento de los clientes, pero no se hace monitoreo de las necesidades de los clientes.
19	1,2,3	<b>*Completo,</b> Las fallas siempre son atendidas y gestionadas por el responsable del proceso/componente.	Ago., 2011	Identificar los SPoF en la infraestructura, plataforma, servicios y procesos para evitar fallas en los servicios.	Cuando se detectan fallas, estas son atendidas por el responsable del área o proceso donde ocurren.
20	3	No	-	-	En plataformas de Cloud, se contratan servidores en diferentes puntos y se separan los componentes syslog del servidor donde se encuentra la aplicación, además de que el riesgo se transfiere al proveedor de servicios de Cloud.
21	2	No	-	Fácil gestión de los servidores al estar enfocados en realizar solo las tareas necesarias para la aplicación.	Son usadas dependiendo de la aplicación, actualmente no han sido utilizadas por que se utilizan servidores dedicados para un propósito específico o aplicación.
22	1,2	<b>*Completo,</b> Los permisos son configurados de acuerdo a lo necesario requerido por los usuarios.	Ene, 2011	Incremento en la seguridad de la información, y permiten a los usuarios tener acceso solo a la información que necesitan.	Los usuarios tienen acceso a sesiones con roles específicos de acuerdo al rol que lleva la persona en la organización.
23	1,2	No	-	-	No ha sido necesario, este tipo de control se hace sobre el tipo de requerimientos del proyecto.

ID	Problema	¿Es implementada?	¿Desde cuándo?	¿Existe algún beneficio?	Como se implementa Actualmente
24	1,2,3	<b>*Parcialmente,</b> Se cuenta con un proceso básico de atención, pero no se tiene establecido formalmente en la organización, la atención se hace sin distinguir o establecer los tipos de problemas relacionados.	Ago., 2011	Designación de responsabilidades en el equipo de desarrollo.	Actualmente se designa un responsable del equipo que haya participado en el proyecto para atender el servicio de mantenimiento. El responsable es supervisado por el Líder(Programador Sénior)
25	1	<b>*Completo,</b> Al hacer las pruebas en un ambiente igual al de producción se garantiza que cuando se actualizan los sistemas se hace de forma rápida evitando las posibilidades de que ocurran fallas.	Ago., 2011	Permitir un rápido paso de las actualizaciones al ambiente de producción.	Se establece un ambiente de pruebas en el que se monta todo el ambiente de desarrollo para probar la nueva actualización.
26	1,2,3	No	-	-	Se establece un proceso de monitoreo sobre los elementos que integran el servicio.
27	1,2,3	<b>*Completo,</b> Al hacer las pruebas en un ambiente igual al de producción se garantiza que cuando se actualizan los sistemas se hace de forma correcta evitando las posibilidades de que ocurran fallas.	Ago., 2011	Aseguramiento de la integración de los nuevos elementos, sin arriesgar el entorno de producción.	Se establece un ambiente de pruebas en el que se monta todo el ambiente de desarrollo para probar la nueva actualización. El flujo de las aplicaciones se diagrama en bizagi lo que permite identificar el impacto y relación con otros componentes desarrollados.



**ANEXO 3 BENCHMARKING DE HERRAMIENTAS DE SOPORTE**

En la siguiente tabla, se muestran las características principales de las herramientas “Help Desk” que fueron analizadas. Como resultado la herramienta OTRS fue la seleccionada para integrar a la propuesta debido a que integra también la extensión de encuestas, está alineado a ITIL y porque es una herramienta que la empresa ya está analizando para integrar en sus procesos actuales.

Herramienta	Costo/licencia	Sistema de Tickets	Web	Reportes	Encuesta	Soporte ITIL
Zendesk <sup>1</sup>	Regular: \$ 24 month Plus: \$ 49 month Enterprise: \$ 99 month	Si	Si	Si	X	X
OTRS <sup>2</sup>	Open Source tool, service and training costs, also is ofered as a software as a service.	Si	Si	Si	Si	Si
RT:Request Traquer <sup>3</sup>	Free app, service and training costs, also is ofered as a software as a service. \$ 100 usd per month, cheapest service plan.	Si	Si	Si	X	X
FreshDesk <sup>4</sup>	3 agents FREE Forever on the Sprouth Plan.	Si	Si	Si	Si	X
Hesk <sup>5</sup>	Free whit "Powered By" links. 39.95 Single Web Site. 199.95 Developer License.	Si	Si	Si	X	X
H2desk <sup>6</sup>	19.99 Month, Cloud Hosted. 399.99 Self Hosted Version.	Si	Si	Si	X	X
IT Help Desk Management <sup>7</sup>	Free up to 5 users.	Si	Si	Si	Si	Si

<sup>1</sup> Zendesk, Consultado el 19 de junio de 2013, <http://www.zendesk.com>

<sup>2</sup> OTRS, Consultado el 16 de Junio de 2013, <http://www.otrs.com/es/>

<sup>3</sup> RT Request Tracker, Consultado el 23 de Junio de 2013, <http://bestpractical.com/rt/>

<sup>4</sup> FreshDesk, Consultado el 24 de Junio de 2013, <http://bestpractical.com/rt/>

<sup>5</sup> Hesk, Consultado el 24 de Junio de 2013, <http://www.hesk.com/>

<sup>6</sup> H2Desk, Consultado el 24 de Junio de 2013, <http://www.h2desk.com/>

<sup>7</sup> IT Help Desk Management, Consultado el 25 de Junio de 2013, <http://www.manageengine.com/help-desk-software.html>

## ANEXO 4 RESULTADOS OBTENIDOS DE LA APLICACIÓN DEL PROCESO DE INTEGRACIÓN Y MEJORA DE LAS PRÁCTICAS FINALES

TABLA 14 LISTADO MEJORES PRÁCTICAS FINALES

ID	AP	Práctica	Descripción
1	1	Iniciar plan de mejora de calidad y de proceso de mantenimiento	Un programa de mejora de la calidad y del proceso ha sido iniciado en el departamento de TI a nivel organizacional. Los gerentes del área de mantenimiento de software están enterados de este programa y han recibido un entrenamiento de concientización inicial sobre la calidad.
24	1	Diseñar y establecer el funcionamiento del servicio de asistencia/moderador/sopORTE al cliente	Diseñar cuidadosamente la funcionalidad de servicio de asistencia / moderador / soporte al cliente alrededor de una capacidad administrativa limitada y aplicación segregada o acceso.
2	1	Aplicar encuestas	Los resultados de encuestas de mantenimiento de software aplicados a los clientes sobre los productos/servicios son usados para identificar mejoras.
3	1	Usar las observaciones, comentarios y quejas para identificar mejoras	Las observaciones, comentarios y quejas de los usuarios/clientes y los grupos de interfaces son recolectados y usados para identificar mejoras.
14	1	Identificar las VBFs	Identificar las funciones vitales del negocio o VBFs (Vital Business Functions).
4	1	Listar y priorizar las mejoras identificadas	Se realiza una lista con las mejoras candidatas a implementarse ordenadas por prioridades, y son usadas como guía para desarrollar el programa de mejora del mantenimiento de software.
5	2	Definir la descripción del proceso de mantenimiento/servicio	Exista al menos una descripción del proceso de mantenimiento/servicio que es usado por el cliente y es ofrecido por la organización.
17	2	<b>+Planificar el Mantenimiento preventivo.</b>	Mantenimiento planificado y preventivo.
16	2	Planear e Integrar al proceso pruebas de disponibilidad.	Programación de pruebas de disponibilidad.
26	2	Planificar periodos de revisión en busca de nuevos métodos o funcionamiento.	Mantenerse con regularidad para identificar nuevas amenazas que afecten la disponibilidad del servicio.
6	3	Planificar capacitaciones referentes al proceso de mantenimiento	El ingeniero de mantenimiento de software es entrenado y motivado a realizar bien las cosas cuando usa el proceso/servicio y ejecuta su rol.
7	3	Planear los tiempos de capacitación	El tiempo de capacitación es planeado.
8	4	Identificar procesos de mantenimiento clave y aplicarles métricas	Algunos procesos de mantenimiento clave, servicios y productos son medidos. Estos son definidos y usados localmente.
9	4	Establecer objetivos de rendimiento y calidad	La organización de mantenimiento ha establecido algunos objetivos de rendimiento y calidad.
		<b>+Realizar Análisis de Rendimiento</b>	
13	4	Realizar análisis de ausencia de disponibilidad	Análisis de ausencia de disponibilidad.
15	4	Realizar análisis de SPoF	Análisis de punto único de fallo o SPoF (Single Point of Failure).
19	4	Prueba de fallas - gestión de fallas	Prueba de fallas – no deje sin gestionar las fallas.
		<b>+Realizar Análisis de Configuración</b>	
21	4	Configurar jaulas "chroot" en plataformas Unix	Usar jaulas "chroot" en plataformas Unix.
22	4	Configurar el mínimo de permisos en sistema de ficheros en todas las plataformas	Usar los mínimos permisos en el sistema de ficheros de todas las plataformas.
23	4	Si es posible, usar sistemas de fichero de solo lectura	Considerar la utilización de sistemas de fichero de solo lectura (como CD-ROM o llaves USB bloqueadas) si fuera posible.
25	4	Parchear rápidamente los sistemas.	Es responsabilidad como desarrollador garantizar que el usuario este tan seguro como sea posible y animar a parchear rápidamente los sistemas vulnerables para garantizar que sus parches sean integrales (es decir, no más parches de este tipo), regresión de problemas anteriores (es decir, soluciones permanecen solventadas), y estable (es decir, que se han realizado suficientes ensayos).
27	4	Realizar actualizaciones de forma segura	Las actualizaciones deben ser prestadas de forma segura, ya sea por firma digital de los paquetes, o mediante un resumen de mensaje libre de colisiones.
10-18	5	<b>+ Establecer revisiones e identificar e investigar novedades y mejoras potenciales en el uso de proceso de mantenimiento o producto.</b>	Nuevos procesos, servicios, tecnologías, metodologías y herramientas son identificados e investigados por su potencial uso en el mantenimiento de software. Considerar también revisar continuamente las necesidades de los clientes para definir mejoras al proyecto o proceso.
20	5	Usar los logs (con copias de solo lectura) para identificar problemas y buscar mejoras.	Genere reportes y realice búsquedas en los logs utilizando una copia de solo lectura o una copia íntegra del original.
11	5	Integrar los nuevos hallazgos* de las investigaciones en el nivel requerido.	Nuevos procesos, servicios, tecnologías, metodologías y herramientas son alcanzados e introducidos en el nivel requerido.
12	5	Las mejoras a los procesos actuales, son iniciados de forma contenida localmente	Mejoras a algunos procesos de mantenimiento, servicios, tecnologías, metodologías y herramientas han sido iniciados de una forma controlada localmente.

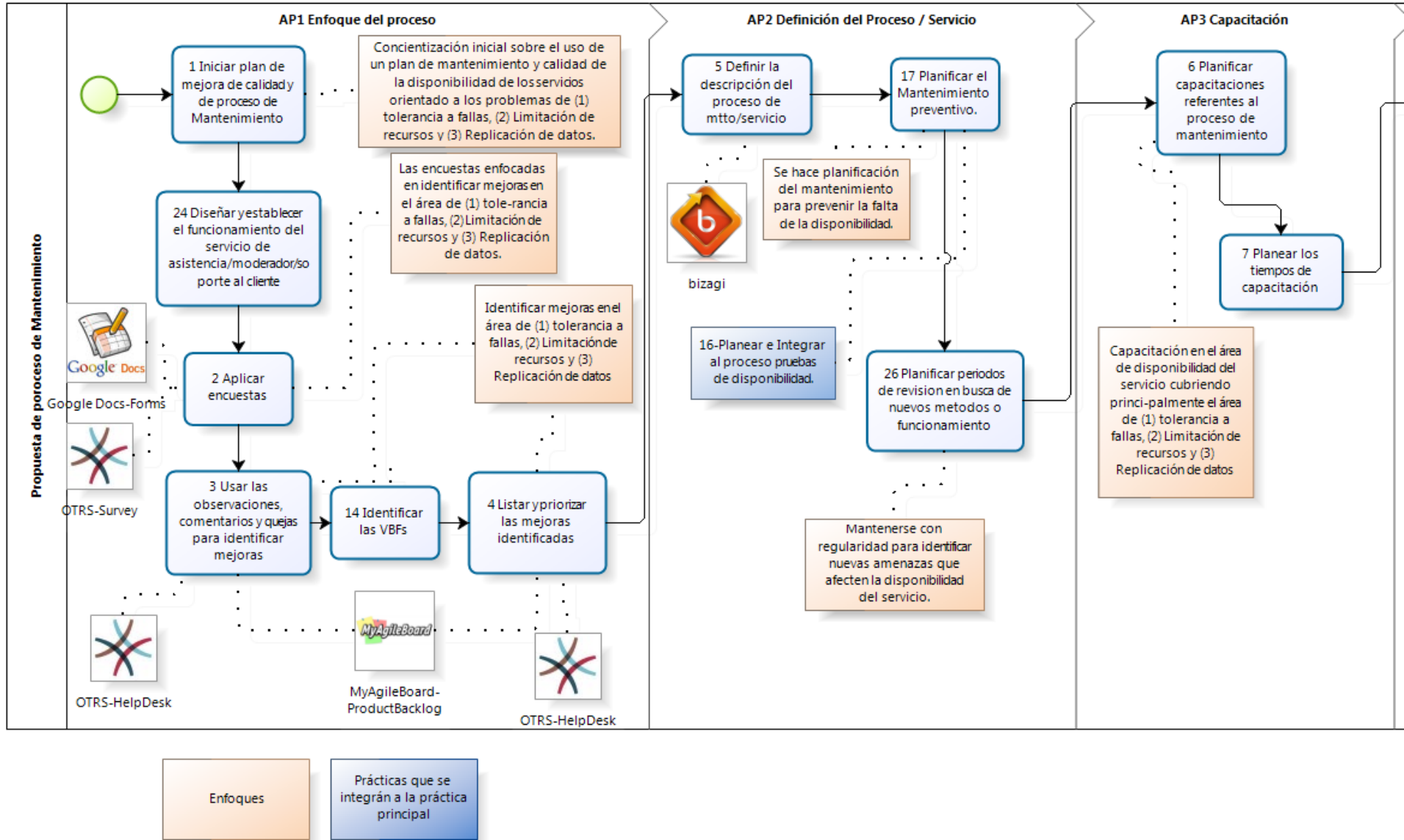


Fig. 7 Secuencia, relación e interacción entre las prácticas, enfoques y herramientas finales. (Parte 1/2)

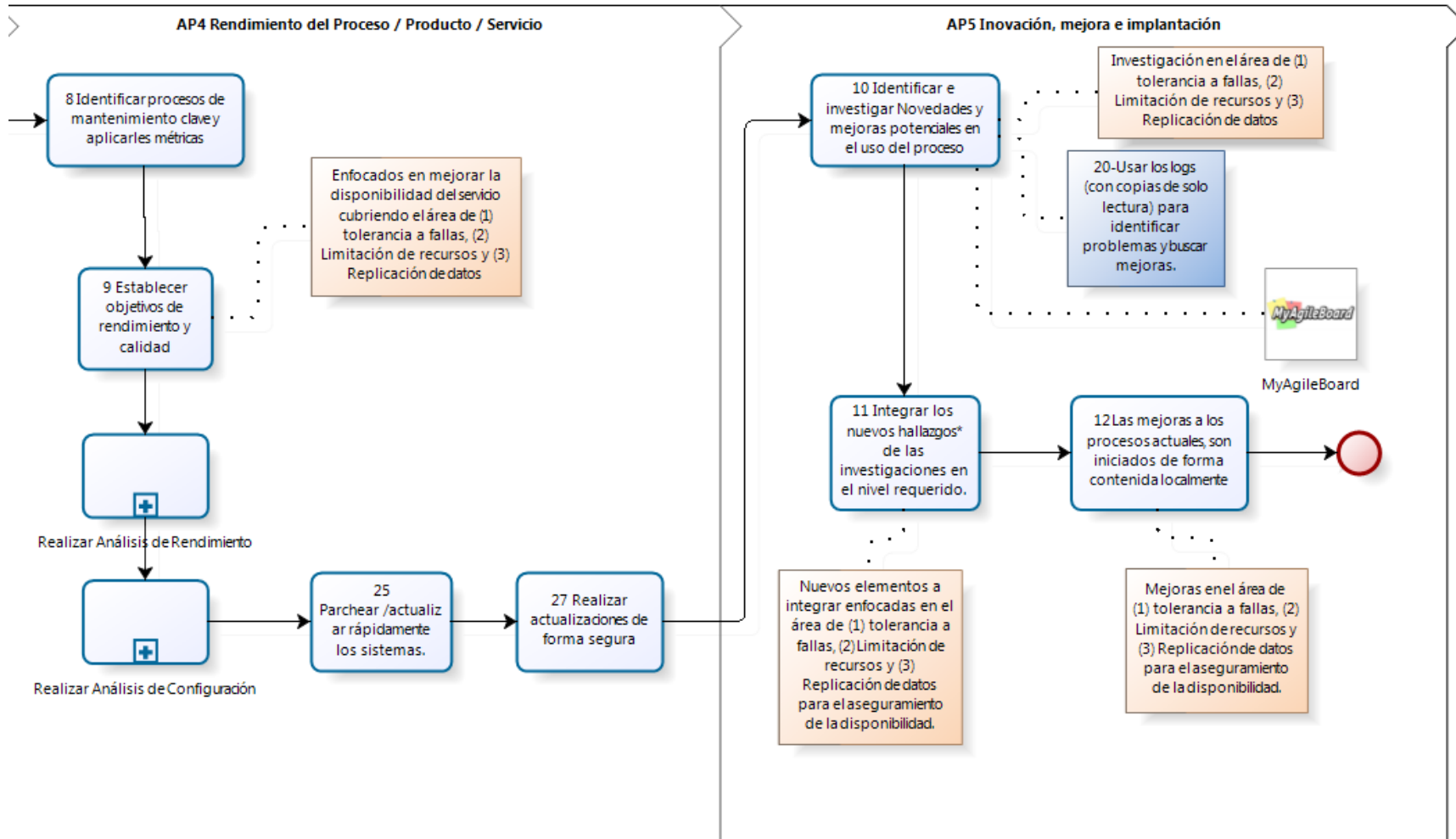


Fig. 8 Secuencia, relación e interacción entre las prácticas, enfoques y herramientas finales. (Parte 2/2)

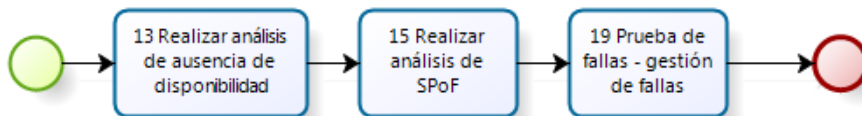


Fig. 9 Subprocesos de la actividad Realizar análisis de Rendimiento

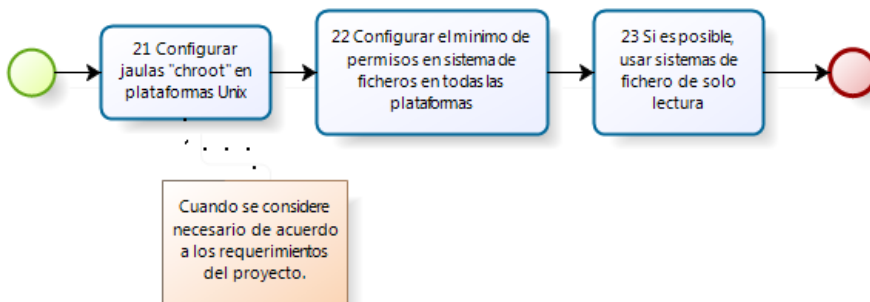


Fig. 10 Subprocesos de la actividad Realizar análisis de Configuración