



**FACULTAD DE INFORMÁTICA
UNIVERSIDAD NACIONAL DE LA PLATA**

**ARQUITECTURA DE UN DATA CENTER CON
HERRAMIENTAS DEVOPS**

**TESIS PRESENTADA PARA OBTENER EL GRADO DE
DOCTOR EN CIENCIAS INFORMÁTICAS**

**Autor: Alberto Belalcázar
Director: Lcdo. Javier Díaz**

Junio 2017

DEDICATORIA

A mis seres queridos gracias por el apoyo incondicional.

A Javier Díaz por su valioso asesoramiento.

A Lía Molinari por sus sugerencias y orientadores conocimientos.

Agradecimientos

A la EEQ y UCE por haberme brindado el apoyo para acceder al doctorado que me ha permitido adquirir conocimientos y experiencias valiosas.

Al personal del Laboratorio de Investigación de Nuevas Tecnologías Informáticas LINTI de la Universidad Nacional de la Plata, en especial a Christian Rodríguez, cuya experiencia en DevOps fue un gran aporte para esta tesis.

Al personal de la secretaría de Posgrado de la Facultad de Informática de la Universidad Nacional de la Plata, por su imperecedero trabajo colaborativo y asistencia oportuna.

A Javier Díaz y Lía Molinari por dirigirme y orientarme en la realización de esta tesis.

RESUMEN

Esta tesis analiza y propone un modelo orientado a la administración eficiente de recursos de un data center como resultado del alineamiento estratégico entre Negocio y Tecnología Informática (de ahora en más, TI), usando un modelo de alineación estratégica y metodologías de desarrollo ágiles, integrando desarrollo y despliegue¹ (deployment) y soportado por buenas prácticas reconocidas internacionalmente.

Con el objetivo de alinear las estrategias del Gobierno Corporativo, del Gobierno de negocio y el Gobierno de TI, se propone un marco conceptual desde el punto de vista heurístico concerniente a la alineación de los procesos Corporativos con TI, aplicando los principios de metodología *Lean Startup* y utilizando y modificando el *Strategic Alignment Model* (SAM) de Henderson y Venkatraman, para lograr una cadena de valor implementada mediante requerimientos y servicios (SAM-RS).

La alineación estratégica corporativa se integra con el modelo colaborativo de DevOps, mediante la incorporación de herramientas al ciclo de vida DevOps, en base a los criterios de buenas prácticas ITIL de disponibilidad, monitoreo, integración continua y versionamiento de las aplicaciones.

El aporte de este trabajo es la integración de gobierno y gestión, considerando mecanismos ágiles de desarrollo y despliegue de aplicaciones mediante un alineamiento estratégico. La adopción de buenas prácticas de reconocimiento internacional fortalece este alineamiento y posiciona a la organización que lo aplique en un lugar destacado hacia la eficiencia de la administración de sus recursos.

Se plantean dos casos de estudio con el objetivo de visualizar la aplicación de los modelos propuestos.

El caso de estudio 1, desde el ámbito de SAM-RS, se evidencia cómo un requerimiento corporativo se transforma en varios requerimientos de TI, y cómo los requerimientos de TI son desarrollados y desplegados al usuario, logrando obtener resultados de acuerdo al marco de los objetivos estratégicos.

En el caso de estudio 2 se aplican las herramientas DevOps de configuración y monitoreo para detectar fallos y restaurar servicios de manera automática en aplicaciones del usuario, con la finalidad de demostrar cómo se logra el alineamiento directo entre la organización y los servicios de software de TI en el marco de las buenas prácticas de disponibilidad.

¹ Deployment: Consiste en la preparación del proceso de entrega del producto al usuario. Incluye la pre-configuración de componentes computacionales, la puesta en producción del producto y aceptación por parte del usuario, y el control de pos-producción.

INDICE DE CONTENIDOS

Contenido

INDICE DE FIGURAS	10
INDICE DE TABLAS	12
1. INTRODUCCIÓN	13
1.1. Motivación	13
1.2. Objetivo General y Específicos	14
1.3. Alcance	15
1.4. Principales Aportes	16
1.5. Enfoque Metodológico	17
1.6. Estructura de la Tesis	18
2. REVISIÓN CONCEPTUAL DE DEVOPS. UN ANÁLISIS DE INTERFAZ ENTRE COBIT Y SAM	21
2.1. DEVOPS	21
2.1.1. Qué es y qué no es DevOps	21
2.1.2. Principios de DevOps	23
2.1.3. Integración Continua	23
2.1.4. Entrega Continua	24
2.1.5. Despliegue Continuo	24
2.1.6. Antecedentes del uso de Herramientas y buenas prácticas como soporte a DevOps	24
2.2. Justificación de uso de SAM mediante análisis conceptual de procesos de COBIT	25
2.2.1. Introducción	25
2.2.2. Objetivo del análisis de relación entre COBIT y SAM	27
2.2.3. Análisis de procesos de COBIT	27
2.2.4. Principios de COBIT	28
2.2.5. Procesos Habilitadores	29
2.2.6. Procesos de Gobierno	29
2.2.7. Procesos de Gestión	30
2.2.8. Modelo de Alineamiento Estratégico (Strategic Alignment Model. SAM)	33
2.2.9. Alcance del modelo SAM	35
2.2.10. COBIT vs SAM	40
2.2.11. Definición de SAM como modelo de alineamiento estratégico	40
2.2.12. Conclusiones	42
2.3. Metodología Lean Startup (MLS)	43
2.4. Buenas Prácticas con Servicios ITIL	45
3. ALINEAMIENTO ESTRATÉGICO ENTRE GOBIERNO CORPORATIVO Y TI, MEDIANTE INCORPORACIÓN DE REQUERIMIENTOS Y SERVICIOS AL MODELO SAM	49
3.1. Estructura de Requerimientos y Servicios como apoyo sobre Dominios del SAM	50
3.1.1. Perspectivas o Escenarios de Alineamiento Corporativo y TI con SAM	52
3.1.2. Perspectivas o Escenarios de Alineamiento (1) y (2)	54
3.1.3. Perspectivas o Escenarios de Alineamiento (3) y (4)	55
3.1.4. Dominios con Estrategias de SAM de acuerdo a Perspectivas	56
3.1.5. Análisis de procesos con Perspectiva de Alineamiento (1) y (2)	56
3.1.6. Análisis de procesos con Perspectiva de Alineamiento (3) y (4)	57
3.2. Inserción de requerimientos y servicios (SAM-RS) por capas en las Perspectivas de Alineamiento con SAM	58
3.2.1. Alineamiento del Proceso 1 con la perspectiva (1) en base a servicios y requerimientos	61
En la Figura 3.11 se presenta el alineamiento del proceso 1 con la perspectiva (1). Es la primera representación de alineamiento que debe ser estructurada en dos partes: PEX – PIN y PIN – GTIN.	61

3.2.2. Alineamiento del Proceso 1 con la perspectiva (2) en base a servicios y requerimientos	62
3.2.3. Alineamiento del Proceso 2 con la perspectiva (3) en base a servicios y requerimientos	63
3.2.4. Alineamiento del Proceso 2 con la perspectiva (4) en base a servicios y requerimientos	64
4. INCORPORACIÓN DE HERRAMIENTAS EN EL MARCO CONCEPTUAL DEL DESARROLLO Y DESPLIEGUE DE APLICACIONES CON DEVOPS	67
4.1. Flujo de tareas o actividades para el caso de nuevas versiones de programas fuentes	67
4.2. Análisis conceptual del Desarrollo de Software	69
4.2.1. Análisis de Roles en base a metodologías ágiles	70
4.2.2. Análisis del Ciclo de vida de las metodologías ágiles	71
4.3. Desarrollo de software en base a DevOps	71
4.3.1. Análisis de Roles en base a DevOps	72
4.3.2. Análisis del Ciclo de vida en base a DevOps	72
4.4. Ciclo de vida en base a DevOps	72
4.4.1. Definición de Requerimientos Cliente (DRC)	72
4.4.2. Priorización Tareas (PRT)	74
4.4.3. Planificación Iteración (PLI)	74
4.4.4. Desarrollo-Iteración (DEI)	75
4.4.5. Pruebas (PUI)	76
4.4.6. Despliegue de la Iteración-DSI	81
4.4.7. Control en Producción de la Iteración-COP	81
4.5. Metodología Ágil SCRUM como apoyo a DevOps	82
4.6. Componentes del Despliegue de Aplicaciones en componentes del Data Center	84
4.6.1. Componentes del Data Center	84
4.6.2. Granularidad de componentes de software	86
4.6.3. Pre-configuración de Componentes de Data Center	90
4.6.4. Despliegue de aplicaciones en el Data Center	91
4.6.5. Importancia de la identificación de componentes de software del Data Center que utiliza una aplicación	93
4.7. Incorporación de Herramientas DevOps como apoyo al Desarrollo y Despliegue de Aplicaciones	94
4.7.1. Herramientas de Integración Continua	98
4.7.2. Herramientas de Gestión de Versionamiento	102
4.7.3. Herramientas de Configuración	105
4.7.4. Herramientas de Gestión de Virtualización	107
4.7.5. Herramientas de Monitoreo	110
4.7.6. Protocolo de Monitoreo en la barrera de la tolerancia a Fallos	112
5. MODELO DE APOYO Y RELACIÓN ENTRE BUENAS PRÁCTICAS ITIL, MLS y MÉTRICAS CON DEVOPS	117
5.1. Estructura de Servicios ITIL y su relación con DevOps	117
5.2. Servicios de ITIL	117
5.2.1. Soporte del Servicio	118
5.2.2. Entrega del Servicio	121
5.3. Puntos de Alineamiento Estratégico entre ITIL y DevOps	123
5.4. Estructura de relación de principios de MLS con DevOps	125
5.4.1. Relación de MLS y DevOps	125
5.5. Aporte de Métricas a DevOps	126
6. ALINEANDO PLANIFICACIÓN ESTRATÉGICA, REQUERIMIENTOS Y SERVICIOS	133
6.1. Modelo orientado a la administración eficiente de recursos de un Data Center	134
6.2. Relaciones gráficas del enfoque de modelado	134
6.2.1. DevOps – MLS	134
6.2.2. Relación DevOps - ITIL	135
6.2.3. Relación MLS - SAM-RS	136
6.2.4. Relación ITIL – Infraestructura interna de TI – Estrategia externa de TI	137
6.3. Hoja de ruta del modelo orientado a la administración eficiente de recursos de un Data Center	137
6.4. Caso de Estudio 1: Alineamiento Corporativo con TI	140
6.4.1. Determinar si una empresa tiene PE y GTI	140

6.4.2. Determinar ámbito de trabajo y necesidades estratégicas de la organización	141
6.4.3. Definir personal administrativo representante de la alta gerencia para llevará adelante la implantación del PE	144
6.4.4. Definir personal de TI representante de TI para llevar adelante el proceso de implantación de GTI	144
6.4.5. Determinar principios de MLS que sirvan de apoyo a las perspectivas	145
Los principios de MLS a tomar en cuenta serán basados en la estructura que se presente en las diferentes perspectivas. El principio de pivotaje servirá de guía en la búsqueda de requerimientos del usuario. Los principios de Idear, Innovar, Aprender, Optimizar generan espacios de conocimiento para definir procesos acorde a las necesidades de los usuarios y ayudan a alinear los objetivos corporativos con los recursos de TI.	145
6.4.6. Socializar y Capacitar el PE y GTI a personal administrativo y TI	145
6.4.7. Capacitar a personal administrativo y TI en modelo de alineamiento estratégico y buenas prácticas	145
6.4.8. Describir las estrategias externas e internas de la organización	145
6.4.9. Determinar Requerimientos externos e internos de la organización	146
6.4.10. Determinar Servicios externos e internos corporativos y TI	147
6.4.11. Problemas al iniciar la implementación de SAM-RS.....	148
6.5. Aplicar proceso de Alineamiento (1) Perspectiva (1) y (2).....	148
6.5.1. Proceso de Alineamiento (1) Perspectiva (1) PEX-PIN.....	149
6.5.2. Proceso de Alineamiento (1) Perspectiva (1) PIN'-GTIN	150
6.5.3. Proceso de Alineamiento (1) Perspectiva (2) PEX-GTIEX-GTIN'	151
6.5.4. Proceso de Alineamiento (1) Perspectiva (2) PEX-GTIEX	152
6.5.5. Proceso de Alineamiento (1) Perspectiva (2) GTIEX - GTIN'	152
6.6. Aplicar proceso de Alineamiento (2) Perspectiva (3) y (4).....	154
6.6.1. Proceso de Alineamiento (2) Perspectiva (3) GTIEX-PEX	154
6.6.2. Proceso de Alineamiento (2) Perspectiva (3) PEX' - PIN'	155
6.7. Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN'' - PIN''	156
6.7.1. Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN''	157
6.7.2. Proceso de Alineamiento (2) Perspectiva (4) GTIN'' - PIN''	158
6.7.3. Análisis de Requerimientos alineados entre Dominios de SAM.....	159
6.8. Desarrollo de aplicaciones con herramientas DevOps en base a requerimientos definidos	161
6.8.2. Evaluación del modelo de alineamiento SAM-RS.....	172
6.8.3. Análisis de encuesta para evaluación del modelo de alineamiento SAM-RS.	173
6.8.4. Evaluación de las herramientas DevOps.....	175
6.8.5. Conclusiones de la aplicación de herramientas DevOps.....	176
6.8.6. Resultados obtenidos del desarrollo de software del caso de estudio 1	177
6.8.7. Conclusión del Caso de Estudio 1.....	179
6.9. Caso de Estudio 2	180
6.9.1. Protocolo para control de software en producción.....	180
6.10. Conclusión de los dos Casos de Estudio	191
7. CONCLUSIONES Y TRABAJOS FUTUROS	192
7.1. Conclusiones.....	192
7.2. Trabajos Futuros	194
8. BIBLIOGRAFÍA	195

INDICE DE FIGURAS

Figura 1.1 Metodología empleada en la realización de esta Tesis	17
Figura 2.1 Estructura del Gobierno Empresarial.....	26
Figura 2.2 Modelo de Aleación Estratégica (Henderson & Venkatraman, 1990)	34
Figura 2.3 Interfaz entre Dominios de SAM con Procesos de COBIT	41
Figura 2.4 Propuesta de modelos, y metodologías, y buenas prácticas a ser estudiadas como soporte a DevOps.....	43
Figura 2.5 Diagrama de Principios de la Metodología Lean Startup.....	45
Figura 3.1 Ajustes de alineación entre tres dominios vinculados secuencialmente.....	49
Figura 3.2 Capa de gestión de dominios en base a requerimientos y servicios	51
Figura 3.3 Perspectivas de relación de estrategias de los dominios de SAM	53
Figura 3.4 Mapa de Dominios con el flujo de las Perspectivas	54
Figura 3.5 Perspectivas de Alineamiento con procesos (1) y (2)	54
Figura 3.6 Perspectivas de Alineamiento con procesos (3) y (4)	55
Figura 3.7 Mapa Dominios con Estrategias de SAM de acuerdo a Perspectivas	56
Figura 3.8 Bloque de Dominios y perspectivas (1) y (2).....	57
Figura 3.9 Bloque de Dominios y perspectivas (3) y (4).....	58
Figura 3.10 Inserción de Servicios y Requerimientos en el Mapa Dominios con Estrategias de SAM (SAM-RS)	59
Figura 3.11 Bloque de Dominios y proceso 1 con perspectiva (1) para inserción de servicios y requerimientos	61
Figura 3.12 Bloque de Dominios y proceso 1 con perspectiva (2) para inserción de servicios y requerimientos	62
Figura 3.13 Bloque de Dominios y proceso 2 con perspectiva (3) para inserción de servicios y requerimientos	63
Figura 3.14 Bloque de Dominios y proceso 2 con perspectiva (4) para inserción de servicios y requerimientos	64
Figura 4.1 Conceptos relacionados de la Metodología de Software Ágil	70
Figura 4.2 Zona Oscura de un Proyecto	74
Figura 4.3 Pruebas Unitarias en Desarrollo de Software Ágil	77
Figura 4.4 Pruebas de Integración Descendentes en Profundidad.....	78
Figura 4.5 Pruebas de Integración Descendentes en Anchura	79
Figura 4.6 Visualización de componentes DevOps en el inicio de desarrollo de software.....	82
Figura 4.7 Roles de los integrantes de una metodología ágil en base a SCRUM	83
Figura 4.8 Componentes de un Data Center.....	85
Figura 4.9 Flujo de generación de servicios hasta el usuario.....	86
Figura 4.10 Provisión de Servicios del Data Center	88
Figura 4.11 Granularidad de Servicio del sistema operativo	89
Figura 4.12 Granularidad de Servicio del sistema operativo	90
Figura 4.13 Despliegue de Iteración	92
Figura 4.14 Incorporación de herramientas en el ciclo de vida DevOps.....	95
Figura 4.15 Incorporación de herramientas DevOps a componentes del desarrollo y despliegue de aplicaciones con DevOps	98
Figura 4.16 La figura muestra como Git almacena la información como instantáneas a lo largo del desarrollo y despliegue de las aplicaciones.	104
Figura 4.17 Secuencia de procesos de Git y GitHub	105
Figura 4.18 Estructura de Vagrant	109
Figura 4.19 Tiempos de inactividad y latencia en tolerancia a fallos	113
Figura 4.20 Tiempos permisibles para detección y control de fallos en forma automática	114
Figura 5.1 Alcance del Ciclo de Vida de DevOps	117
Figura 5.2 Ambientes de influencia de DevOps e ITIL.....	124
Figura 5.3 Relación entre componentes DevOps e ITIL	124
Figura 5.4 Pirámide de Métricas de DevOps (Fuente Gartner 2014).....	128
Figura 6.1 Estructura de componentes para un manejo eficiente de recursos de un data center	134
Figura 6.2 Bloque de Dominios Perspectiva (1) PEX-PIN	149

Figura 6.3 Establecimiento de nueva estrategia de apoyo interno PIN03.....	150
Figura 6.4 Bloque de Dominios Perspectiva (1) PIN'-GTIN	150
Figura 6.5 Establecimiento de nueva estrategia GTIN02 para Gobierno Interno de TI	151
Figura 6.6 Bloque de Dominios con proceso 1 con perspectiva (2) con GTIN modificado	151
Figura 6.7 Bloque de Dominios Perspectiva (2) PEX-GTIEIX	152
Figura 6.8 Bloque de Dominios Perspectiva (2) GTIEIX-GTIN'	152
Figura 6.9 Establecimiento de nueva estrategia GTIN03 para Gobierno Interno de TI	153
Figura 6.10 Proceso de Alineamiento (2) Perspectiva (3)-(4) GTIEIX – PEX - PIN' y GTIEIX - GTIN'' - PIN'	153
Figura 6.11 Bloque de Dominios Proceso de Alineamiento (2) Perspectiva (3) GTIEIX-PEX-PIN' ..	154
Figura 6.12 Proceso de Alineamiento (2) Perspectiva (3) GTIEIX-PEX	154
Figura 6.13 Establecimiento de nuevas estrategias luego de analizar GTIEIX' sobre PEX	155
Figura 6.14 Proceso de Alineamiento (2) Perspectiva (3) PEX' - PIN'	155
Figura 6.15 Establecimiento de nuevas estrategias luego de analizar PEX' sobre PIN'	156
Figura 6.16 Proceso de Alineamiento (2) Perspectiva (4) GTIEIX' - GTIN'' - PIN''	156
Figura 6.17 Proceso de Alineamiento (2) Perspectiva (4) GTIEIX' - GTIN''	157
Figura 6.18 Establecimiento de nuevas estrategias luego de analizar GTIEIX' sobre GTIN''	157
Figura 6.19 Proceso de Alineamiento (2) Perspectiva (4) GTIN''' - PIN''	158
Figura 6.20 Establecimiento de nuevas estrategias luego de analizar GTIN'' sobre PIN''	158
Figura 6.21 Cronograma de actividades por tareas	165
Figura 6.22 Tiempos de Iteración 1	167
Figura 6.23 Tiempos de Iteración 2	168
Figura 6.24 Campos de la aplicación.....	169
Figura 6.25 Tiempos de la Iteración 3.....	170
Figura 6.26 Mensaje de facturada cancelada directo al usuario de la aplicación	171
Figura 6.27 Tiempos de Iteración 4	171
Figura 6.28 Tiempos de Iteración 5	172
Figura 6.29 Tiempos planificados vs tiempo real.....	178
Figura 6.30 Control de monitoreo sobre componentes del Data Center	181
Figura 6.31 Monitoreo de servicios de Apache con Nagios por componentes de sistema operativo	187
Figura 6.32 Identificación de Tiempos Críticos durante el error	188
Figura 6.33 Resultado de análisis de error.....	188
Figura 6.34 Código de restauración de servicio con Puppet.....	189
Figura 6.35 Restauración de servicio con Puppet con tiempos críticos luego de un error	190

INDICE DE TABLAS

Tabla 2-1 Cuatro perspectivas dominantes sobre la planificación de TI.....	39
Tabla 2-2 Ciclo de vida de los Servicios de ITIL	46
Tabla 2-3 Capa de procesos ITIL para efectivizar la gestión del desarrollo y despliegue con DevOps	47
Tabla 3-1 Procesos y perspectivas de alineamiento	56
Tabla 3-2 Entradas y Salidas de Estrategias de acuerdo a perspectivas (1) y (2).....	57
Tabla 3-3 Entradas y Salidas de Estrategias de acuerdo a perspectivas (3) y (4).....	58
Tabla 3-4 Entradas y Salidas de Estrategias de proceso 1 y perspectiva (1).....	61
Tabla 3-5 Entradas y Salidas de Estrategias de proceso 1 y perspectiva (2).....	62
Tabla 3-6 Entradas y Salidas de Estrategias de proceso 2 y perspectiva (3).....	64
Tabla 3-7 Entradas y Salidas de Estrategias de proceso 2 y perspectiva (4).....	65
Tabla 4-1 Ciclo de Vida de las Metodologías Ágiles	71
Tabla 4-2 Servicios activos de Software de Data Center	87
Tabla 4-3 Recursos y servicios que administran desarrolladores y operadores	96
Tabla 4-4 Entornos para la correcta administración de desarrollo y despliegue de aplicaciones ..	96
Tabla 5-1 Servicios en el ciclo de vida de DevOps.....	121
Tabla 6-1 Resultados evaluación 30 empresas	140
Tabla 6-2 Número de cortes de energía promedio mensual por falta de pago.....	143
Tabla 6-3 Entradas y Salidas de estrategias externas e internas del negocio y TI	146
Tabla 6-4 Requerimientos Externos e Internos del Negocio y TI	147
Tabla 6-5 Servicios Externos e Internos del Negocio y TI.....	147
Tabla 6-6 Alineación PEX-PIN Original	149
Tabla 6-7 Actualizada Estrategia nueva PIN03 con generación de PIN'	150
Tabla 6-8 PIN' - GTIN original.....	151
Tabla 6-9 Estrategia actualizada para GTIN'	151
Tabla 6-10 PEX y GTIEX sin variación	152
Tabla 6-11 GTIEX y GTIN' con servicios y requerimientos antes de análisis de GTIEX.....	152
Tabla 6-12 GTIEX y GTIN' con su nueva estrategia GTIN03	153
Tabla 6-13 Influencia de GTIEX sobre PEX resultando PEX'	155
Tabla 6-14 Servicios y requerimientos nuevos luego análisis de PEX' sobre PIN'	156
Tabla 6-15 Servicios y requerimientos nuevos luego de análisis de GTIEX' sobre GTIN''	158
Tabla 6-16 Nuevos servicios y requerimientos luego de analizar GTIN''' sobre PIN''	159
Tabla 6-17 Resumen Requerimientos Externos Corporativos	159
Tabla 6-18 Resumen Requerimientos Internos Corporativos.....	159
Tabla 6-19 Consolidación Requerimientos Externos de GTI	160
Tabla 6-20 Consolidación de Requerimientos Internos del GTI.....	160
Tabla 6-21 Flujo de tareas para el equipo de DevOps	161
Tabla 6-22 Tiempos y frecuencia de tareas por responsable	164
Tabla 6-23 Herramientas DevOps usadas en las aplicaciones por tarea	164
Tabla 6-24 Encuesta para evaluar Plan estratégico y modelo SAM-RS.....	173
Tabla 6-25 . Control del ciclo de vida para el equipo DevOps	175
Tabla 6-26 Recuperación de valores por disminución en tiempos de reconexión	179
Tabla 6-27 Procesos relacionados con la entrega del servicio de sistema operativo	184
Tabla 6-28 Procesos relacionados con la entrega del servicio de sistema operativo	185
Tabla 6-29 Servicios activos del sistema operativo	185
Tabla 6-30 Servicios activos del sistema operativo Linux	186
Tabla 6-31 Servicios activos de Apache	186
Tabla 6-32 Tiempos de duración desde el error hasta su identificación y envío de control	188
Tabla 6-33 Tiempos de duración desde la toma de control hasta su restauración con Puppet ..	189

1. INTRODUCCIÓN

1.1. Motivación

En los últimos años, los servicios que ofrecen las áreas de tecnología de la información y las comunicaciones (TIC) de las organizaciones, se han incorporado como un componente de uso común dentro de la vida cotidiana.

El uso inteligente de las TIC se ha convertido en el facilitador de la transformación notable en las organizaciones (Toomey, 2009). Estas han tenido que replantear su modelo de negocio ante la demanda de un usuario cada vez más habituado y exigente en el uso de la tecnología, por ejemplo, para acceder a los servicios en forma remota, tanto para quienes están geográficamente alejados de los puntos de atención, como quienes no lo están.

La exigencia planteada por los usuarios en cuanto a obtener mejores servicios tecnológicos, genera en las organizaciones un cambio de paradigma que debe ser sostenido por estrategias empresariales. La brecha entre estas estrategias, los objetivos corporativos y la administración de los servicios de TI puede ser difícil de alinear sin una planificación y compromiso claramente establecido.

Las TIC deben ofrecer prestaciones estratégicas e innovadoras que den valor agregado a los requerimientos corporativos.

El valor agregado se consigue mediante la gestión integrada de las tareas que componen la cadena de valor de la empresa en forma conjunta con la cadena de proveedores y clientes (Ordiz & Pérez, 2002).

COBIT 5 define a los recursos tecnológicos como el conjunto de aplicaciones, datos, infraestructura y recursos humanos (ISACA, 2017). En este contexto de aprovechamiento ante las transformaciones que permiten las TIC, las organizaciones deben abordar la definición y administración adecuada de estos recursos para que al estar alineados con los objetivos se transformen en facilitadores en la creación de valor. Las corporaciones exitosas se distinguen por su capacidad para aprovechar los recursos en los procesos de transformación de sus organizaciones (Henderson & Venkatraman, 1990).

Sin esquemas de alineamiento directo con los objetivos empresariales se desperdicia el gran potencial de TI y se incurre en gastos innecesarios.

La estructura organizativa es uno de los puntos clave en este objetivo de alineamiento. Por ello se hace necesario identificar todos sus componentes con un enfoque orientado a procesos, basado en funciones y en las capacidades adecuadas de los distintos actores de acuerdo a sus responsabilidades, es decir, determinar claramente los roles .

Tomemos por caso las áreas de desarrollo y operaciones de TI. Tradicionalmente estas áreas están separadas en las estructuras organizativas. Haciendo una pronunciada simplificación, el flujo de trabajo ante el desarrollo de una aplicación transita por las siguientes fases:

- Desarrollo construye o elabora programas y nuevos servicios informáticos.
- Los programas son testeados.
- Se realiza el despliegue de los programas a producción, donde los nuevos servicios son provistos a los usuarios internos y externos. Este paso se lo realiza en forma manual o automática.

Cuando el usuario utiliza estas aplicaciones puede encontrar fallos o resultados no esperados, ya sea a nivel de desempeño o funcionalidad. Ante sus reclamos, por las vías establecidas por la organización (*call center*, envío de mails o llamados telefónicos) el escalamiento de ese incidente puede dificultarse ante la imprecisión acerca de quién debe hacerse cargo. Este retardo en la resolución del fallo perjudica al usuario, pues no logra la respuesta esperada en tiempo y forma. Como consecuencia, esto incide directamente en una percepción negativa de la imagen de la empresa que puede resultar en la pérdida de ese usuario/cliente.

La falta de determinación acerca de las causas del incidente, generan una situación de inestabilidad que en muchos casos se manifiesta en desprestigio acerca de las TICs.

Una de las razones de esta situación es la falta de integración entre las distintas áreas. El problema no son las TICs: el problema es la forma en cómo se planifican y administran.

La detección temprana de los incidentes y una respuesta ágil desde la organización, permitiría reducir el tiempo para solucionar el incidente y cerrarlo para satisfacción del usuario. También sería un aporte a la mejora continua desde el replanteo de situaciones y responsabilidades.

Es por esto que surge la necesidad de contar con elementos metodológicos que permitan alinear la estrategia corporativa con la prestación de servicios, donde las diferentes áreas funcionales estén integradas para dar respuesta a los requerimientos de usuarios.

En este trabajo de tesis se aborda DevOps como una herramienta para la integración de desarrolladores y operadores en un contexto de metodologías ágiles de desarrollo.

1.2. Objetivo General y Específicos

Objetivo General

El objetivo general de la tesis es contribuir a la mejora en la gestión del desarrollo y despliegue de aplicaciones como resultado de un alineamiento estratégico de los servicios Corporativos y Tecnología de la información.

Objetivos Específicos

- Proponer una interfaz entre los componentes de COBIT y SAM a fin de justificar el uso de este, como modelo de alineamiento estratégico.
- Plantear la determinación de un marco conceptual desde el punto de vista heurístico con el objetivo de alinear y mejorar las estrategias del Gobierno Corporativo, del Gobierno de negocio y el Gobierno de TI, mediante la adaptación del *Strategic Alignment Model (SAM)* de Henderson y Venkatraman, dado como resultado SAM-RS, para lograr una cadena de valor implementada mediante requerimientos y servicios.
- Plantear un marco conceptual de DevOps en el que se detallan los componentes del desarrollo y despliegue de aplicaciones, basado en principios de integración, entrega y despliegue continuo, que sirven para efectivizar la labor colaborativa de DevOps.

- Plantear la especificación e incorporación de herramientas DevOps en las fases del ciclo de vida, en base a las necesidades del negocio de disponibilidad, control de gestión o monitoreo, integración continua y versionamiento, para lograr un control efectivo de las aplicaciones.
- Proponer un protocolo para el control de monitoreo de software en producción, a fin de garantizar que las herramientas de monitoreo DevOps consigan disponibilidad de servicios a los usuarios.
- Proponer la incorporación de buenas prácticas que proveen ITIL, los principios de MLS y las Métricas, en cada una de las etapas del ciclo de vida del desarrollo y despliegue de aplicaciones con DevOps, a fin de entregar aplicaciones que aporten valor agregado a las labores diarias del usuario.
- Proponer un modelo para la administración eficiente de recursos de un *Data Center* en forma gráfica y en base a una hoja de ruta, que sirva como guía para la implantación del modelo SAM-RS y la implementación de las herramientas DevOps.
- Contrastar el modelo con dos casos de estudio. El primero, una aplicación del MAS-RS y el segundo la aplicación de las herramientas DevOps en casos reales.

1.3. Alcance

Este trabajo de tesis se enmarca en un análisis conceptual de la metodología SAM, Lean Startup e ITIL como soporte a los componentes del desarrollo y despliegue de aplicaciones basados en DevOps, a fin de lograr un alineamiento efectivo de los recursos de TI con las estrategias de la organización.

Se abordarán las interfaces entre el modelo SAM y COBIT, mediante un análisis conceptual, que permita justificar el uso de un modelo de alineamiento estratégico.

Se propondrá un modelo en base a requerimientos y servicios (SAM-RS) como resultado de la modificación al Modelo SAM de Henderson y Venkatraman, haciendo uso de los conceptos de gobierno corporativo y TI. Se establecerá mecanismos de alineación involucrando distintas perspectivas estratégicas corporativas y TI. Este enfoque ayudará a determinar con más claridad las verdaderas necesidades y requerimientos a ser aplicados en el desarrollo y despliegue de aplicaciones.

Se analizarán los conceptos de ITIL, de la metodología Lean Startup (MLS) y de Métricas que servirán de soporte para la determinación del modelo de relación con el ciclo de vida con DevOps, a fin de obtener alineamiento efectivos de los objetivos empresariales con TI.

Se analizarán los conceptos de DevOps teniendo como soporte a las metodologías ágiles. Se expondrá un enfoque de modelo de incorporación de herramientas DevOps en los componentes del desarrollo y despliegue de aplicaciones con DevOps, a fin de determinar las herramientas DevOps más adecuadas en base a criterios de disponibilidad, control de gestión o monitoreo, integración continua y versionamiento.

Se planteará un análisis conceptual de la gestión de monitoreo sobre las aplicaciones. Se propone protocolo para el control de monitoreo de software en producción, a fin de garantizar que las herramientas de monitoreo DevOps consigan disponibilidad de servicios.

Se presentarán dos casos de estudio, el primero a fin de verificar la aplicabilidad de la adaptación realizada al modelo SAM en base a requerimientos y servicios (SAM-RS), y el segundo para verificar que la aplicación adecuada de las herramientas DevOps brinda un aporte eficaz al alineamiento corporativo y TI.

1.4. Principales Aportes

Los aportes realizados por esta tesis son:

- Aporte conceptual que plantea una interfaz entre los componentes de COBIT y SAM a fin de justificar el uso de este, como modelo de alineamiento estratégico.
- La determinación de un marco conceptual desde el punto de vista heurístico con el objetivo de alinear y mejorar las estrategias del Gobierno Corporativo, del Gobierno de negocio y el Gobierno de TI, mediante el *Strategic Alignment Model* (SAM) de Henderson y Venkatraman.
Con apoyo de los principios de *Lean Startup* (Ries, 2016), se realiza una modificación y adaptación del modelo SAM dando como resultado SAM-RS, para lograr una cadena de valor implementada mediante requerimientos y servicios.
- Aporte conceptual de DevOps enmarcado en principios de integración, entrega y despliegue continuo. Los roles y el ciclo de vida DevOps con sus fases detalladas en componentes de desarrollo y despliegue de software, consiguen organizar las actividades, incorporando agilidad en la entrega de prestaciones al usuario.
- La especificación e incorporación de herramientas DevOps en las fases del ciclo de vida para alcanzar un control efectivo de las aplicaciones, que permitan alinear los recursos del desarrollo y despliegue con los objetivos corporativos.
- Se plantea un protocolo para el control de monitoreo de software en producción, a fin de garantizar que las herramientas de monitoreo DevOps consigan disponibilidad de servicios a los usuarios. De esta manera se logra alineamiento efectivo entre la Organización y TI mediante el uso adecuado de los recursos de TI.
- La incorporación de buenas prácticas que provee ITIL, de los principios de MLS y de Métricas, en cada una de las etapas del ciclo de vida del desarrollo y despliegue de aplicaciones con DevOps, a fin de alcanzar un alineamiento adecuado en todas las fases de desarrollo de software que aporten valor agregado a las labores diarias del usuario.
- Se propone un modelo para la administración eficiente de recursos de un Data Center en forma gráfica y en base a una hoja de ruta, que sirva como guía para

la implantación del modelo SAM-RS y la implementación de las herramientas DevOps.

- Dos casos de estudio. El primero es la aplicación del MAS-RS y el segundo la aplicación de las herramientas DevOps en componentes software de una Data Center. En los dos casos prácticos, se presentan resultados para cada caso.
- Como contraste de estos aportes previos, se han presentado y publicados los siguientes artículos con la participación del autor en Congresos Internacionales reconocidos y revista periódica con referato internacional.
 - Towards the Strategic Alignment of Corporate Services with IT, applying Strategic Alignment Model (SAM).
 - Artículo presentado en The Journal of Computer Science and Technology (JCS&T) Vol. 16 – No. 1 – April 2016. ISSN: 1666-6038 (Online), 1666-6046 (Print). Páginas 1 a 7.
 - Principios, Roles y Métricas en alineamiento estratégico de nuevos requerimientos utilizando DevOps.
 - Artículo presentado en el Congreso Internacional CACIC 2016 Universidad Nacional de San Luis Argentina. ISBN: 978-987-733-072-4. Libro de actas Páginas 1042 – 1051.
 - Control y Recuperación de los componentes de un Data Center con herramientas DevOps-OpsCode aplicando Gestión de ITIL con análisis en la barrera de detección y tolerancia a fallos.
 - Artículo presentado en el Congreso Internacional CONAIISI 2015 realizado en UTN Buenos Aires Argentina. ISBN: 978-987-1896-47-9. Páginas 1 – 8.
 - Towards Complement Strategy in the Business-IT Alignment
 - Artículo presentado en Congreso Internacional Proceedings IEEE CHILECON 2017, realizado en Pucón, Chile. ISSN: 0719-6806. Pág. 632-637.

1.5. Enfoque Metodológico

En la Figura 1.1 se presenta un breve esquema metodológico empleado para la realización de la presente tesis doctoral.

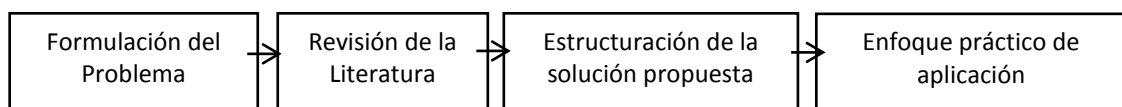


Figura 1.1 Metodología empleada en la realización de esta Tesis

a. Formulación del Problema

- Definición del problema (Sección 1.1)

b. Revisión de la Literatura: Búsqueda

- Selección, recopilación y revisión de trabajos relacionados pertinentes.

c. Estructuración de la metodología propuesta

La metodología propuesta se ha organizado en base a la siguiente secuencia de actividades:

- Aporte conceptual que plantea una interfaz entre los componentes de COBIT y SAM a fin de justificar el uso de este, como modelo de alineamiento estratégico.
- Definición de los principios de la Metodología Startup.
- Alineamiento Estratégico Gobierno Corporativo-Gobierno de TI apoyado en la estructura de requerimientos y servicios realizados al modelo SAM.
- Análisis de los Componentes de Desarrollo y Despliegue de aplicaciones.
- Inserción de herramientas a componentes del ciclo de vida DevOps.
- Inserción de Servicios de ITIL y principios de MLS a los componentes de desarrollo y despliegue de aplicaciones con DevOps, como apoyo en la disponibilidad de servicios Corporativos y TI.
- Definición del modelo para la administración eficiente de recursos de un Data Center en forma gráfica y en base a una hoja de ruta, que sirva como guía para la implantación del modelo SAM-RS y la implementación de las herramientas DevOps.

d. Enfoque práctico de aplicación

- Se presenta dos enfoques de aplicación real.

1.6. Estructura de la Tesis

La estructura de la tesis permite visualizar el contenido de los capítulos que conforman esta tesis.

- Capítulo 1

En este capítulo se define:

- La motivación del estudio
- Se establece el objetivo general y específico del trabajo de tesis
- Se define el alcance de la tesis
- Se definen los principales aportes de la tesis
- Se explica el enfoque metodológico utilizado

- Capítulo 2

Este capítulo aborda la revisión de conceptos de DevOps, a fin de ubicarlo en el contexto empresarial.

Se plantea un análisis de interfaces entre componentes de COBIT y SAM, a fin de justificar la elección de SAM como modelo de alineamiento estratégico entre la organización y TI.

Se aborda el soporte que brinda la Metodología Lean Startup al alineamiento del Gobierno Corporativo y TI, cuyos enfoques y principios interactúan con los principios de DevOps para diseñar nuevos productos en base a criterios empresariales conectados al apoyo de tecnología.

Se realiza un breve análisis del apoyo de los servicios de ITIL al entorno del desarrollo y despliegue con DevOps, en base a Disponibilidad, Monitoreo, Versionamiento de aplicaciones, y como parte de ellas se incluye al principio de DevOps de Integración Continua.

- Capítulo 3

El capítulo 3 aborda una modificación al Modelo SAM de Henderson y Venkatraman, se plantea un contexto de alineamiento en base a requerimientos y servicios desde diferentes perspectivas las cuales son apoyadas por principios de MLS en cada dominio.

- Capítulo 4

El capítulo 4 aborda los componentes de DevOps.

- Componentes de la Gestión de Desarrollo de Software

- Se realiza una granularidad de todos los componentes DevOps que intervienen en el ciclo de vida de desarrollo de aplicaciones con DevOps.

- Componentes de la Gestión de Data Center

- Se analizan los componentes que sirven para el despliegue de aplicaciones y todos los componentes relacionados con las labores de los operadores de sistemas en el Data Center
- Herramientas DevOps como apoyo al Desarrollo y Despliegue de un Proyecto
 - Se detallan las herramientas DevOps que sirven para automatizar las tareas en las labores colaborativas del desarrollo y despliegue de aplicaciones.
 - Se analiza la aplicación de algunas herramientas en base a un caso de estudio donde se visualiza la necesidad de granular los componentes del desarrollo y despliegue de aplicaciones, para un aseguramiento del servicio que se provee al usuario.
 - Se plantea un protocolo para control de software en producción.

- Capítulo 5

En el Capítulo 5 se describen los conceptos de las buenas prácticas ITIL en TI, se presenta conceptos de los siguientes bloques: Soporte del Servicio, Entrega del Servicio, Administración de Infraestructura, Administración de Sistemas, Administración de Seguridad, Administración de Aplicaciones.

Se realiza un análisis sobre las relaciones de ITIL con los conceptos y herramientas DevOps.

Se revisan los conceptos de MLS y se realiza un análisis de congruencia de apoyo en cada etapa del despliegue y desarrollo con DevOps.

El aporte de Métricas al Ciclo de Desarrollo y Despliegue con DevOps, se verifica mediante el triángulo de métricas de Gartner.

- Capítulo 6

El Capítulo 6 aborda la alineación de la planificación estratégica con requerimientos y servicios en base a los siguientes tópicos:

- Se propone un modelo para la administración eficiente de recursos de un Data Center en forma gráfica y en base a una hoja de ruta, que sirve de guía para la implantación del modelo SAM-RS y de la implementación de las herramientas DevOps.
- Se presentan dos casos de estudio con enfoque de aplicación práctica de la vida diaria.

- Capítulo 7

En el Capítulo 7 se trata las conclusiones del trabajo y los estudios futuros relacionados con los temas de este estudio.

2. REVISIÓN CONCEPTUAL DE DEVOPS. UN ANÁLISIS DE INTERFAZ ENTRE COBIT Y SAM

Este capítulo aborda la revisión de conceptos de DevOps, basados en sus principios de integración, entrega y despliegue continuo que ayuda a ubicar DevOps en el contexto empresarial. Se realiza una interfaz entre los componentes de COBIT y SAM a fin de justificar la elección de este como modelo de alineamiento estratégico. Se aborda el soporte que brinda la Metodología Lean Startup al alineamiento del Gobierno Corporativo y TI, cuyos enfoques y principios ayudan a diseñar nuevos productos en base a criterios empresariales conectados al apoyo de tecnología.

Se realiza un breve análisis de los servicios de ITIL como soporte de buenas prácticas en el entorno del desarrollo y despliegue con DevOps, en base a las necesidades del negocio de disponibilidad, monitoreo, versionamiento de aplicaciones, y como parte de ellas se incluye al principio de DevOps de integración continua.

2.1. DEVOPS

DevOps ha surgido como una nueva tendencia orientada hacia la obtención de una relación de trabajo y colaboración entre áreas de desarrollo y operaciones de TIC (Schulze & Adan, 2014).

El objetivo del trabajo colaborativo de DevOps es actualizar y mantener los sistemas de producción estables en todo momento.

Es necesario conocer los alcances en los que se desenvuelve DevOps con el fin de establecer el verdadero ambiente colaborativo que se brinda en el desarrollo y despliegue de aplicaciones.

2.1.1. Qué es y qué no es DevOps

En la actualidad los sistemas informáticos tienen un rol fundamental en la prestación de servicios. A medida que van adquiriendo masividad y protagonismo, se requiere mayor calidad lo que se traduce en seguridad, adaptación a nuevos paradigmas y contextos. Por lo tanto en la instancia de diseño de software se hace necesario la participación de equipos multidisciplinarios para definir un producto que satisfaga a diferentes usuarios. En el desarrollo de un proyecto de software se requiere sincronización entre varios componentes: requerimientos, metodología de desarrollo, desarrolladores/operadores, infraestructura, servicio a los usuarios.

Las estrategias de alineamiento entre desarrolladores y operadores ante requerimientos de desarrollo de aplicaciones, deben ser coordinadas desde las etapas tempranas de planificación del proyecto. Cuando se opta, además por metodologías ágiles de desarrollo estas estrategias deben aprovechar el dinamismo que habilitan este tipo de tecnologías.

Como se indicó en párrafos anteriores, DevOps ha surgido como una nueva tendencia a la integración de las áreas de desarrollo y operaciones de TICs con el objetivo de optimizar la entrega de servicios en un contexto de metodologías ágiles de desarrollo. El objetivo del trabajo colaborativo de DevOps es actualizar y mantener los sistemas de producción estables en todo momento.

DevOps plantea una interacción ágil y colaborativa entre programadores y operadores, desde la mirada tradicional con una segregación marcada de funciones, mediante la

inclusión de mecanismos que le dan un mayor dinamismo a la entrega de los servicios sin descuidar el control desde el inicio del proyecto hasta el control de producción.

Para lograr su objetivo, DevOps se basa en principios tales como Integración Continua, Entrega Continua y Despliegue Continuo.

En los contextos ágiles se generan múltiples integraciones por día, con el aporte de cada miembro del equipo de desarrollo.

Cada integración es automatizada y probada en forma automática, para la detección temprana de errores.

El concepto DevOps está dirigido a la tarea colaborativa de programadores y operadores. A continuación se detallan 4 puntos orientados a la comprensión acerca de qué es y qué no es DevOps (Lent, 2014)(Davis & Daniels, 2015).

1) DevOps no es un puesto de trabajo

La labor colaborativa que tienen los desarrolladores y operadores con DevOps no implica que se deba definir un nuevo rol o puesto de trabajo “DevOps” en las organizaciones. DevOps no debe ser un rol laboral (Lent, 2014).

DevOps se trata de que los desarrolladores entiendan la realidad de las operaciones y de que el equipo de operaciones comprenda lo que involucra el desarrollo.

La concepción de DevOps en los integrantes de los equipos de desarrollo y despliegue de aplicaciones debe ser en base a capacitación y aplicación de buenas prácticas en el día a día de sus labores. No se aplica en un área específica sino que define una nueva modalidad de interrelación entre distintas áreas, incluso la relación con los usuarios. Esto último se hace más evidente ante reclamos por fallos o insatisfacción.

La falta de claridad por parte de los ejecutivos de la organización al solicitar personal que ocupe un puesto administrativo u operativo DevOps es un síntoma que no entienden lo que DevOps es realmente (Lent, 2014).

2) DevOps no es una categoría de herramientas

Las herramientas para el desarrollo y despliegue de software juegan un papel muy importante en el apoyo al ámbito colaborativo de DevOps, pero ello no debe dar lugar a confusiones: no se trata sólo de software. Se trata de cultura (Lent, 2014).

Hay herramientas que soportan esta cultura. Pero las herramientas por sí solas no definen un ambiente DevOps.

3) DevOps no se trata de resolver un problema de TI

Habitualmente hay una manifiesta frustración en los ejecutivos de las organizaciones acerca de los servicios que pueden brindar las TICs, y también es habitual que el personal que administra e implementa las soluciones de las TICs, no logre abstraer los objetivos del negocio (Ross & Weill, 2002). La falta de comunicación entre las áreas de negocio y las de TICs aporta a esta generación de expectativas que luego no se cumplen. Las áreas TICs soportan presiones para implementar o mejorar servicios de diferente complejidad, para resolver tareas operativas o incidentes que exigen rápida solución. Estas emergencias, si bien deben solucionarse, pueden tener como consecuencia que se pierda el alineamiento con los objetivos organizacionales para brindar valor agregado (Belalcázar, Díaz, & Molinari, 2015) (Muñoz & Ulloa, 2011).

La labor de DevOps permite que desarrollo y operaciones colaboren en la entrega de software (Lent, 2014). Aunque no es el objetivo final. El punto de DevOps es permitir a su empresa reaccionar ante las fuerzas del mercado lo más rápido, eficiente y confiable como le sea posible. Para una adecuada implementación de DevOps debe existir alineamiento entre la organización y los servicios que ofrece TI.

4) DevOps no es un sinónimo de Integración Continua

DevOps se propone en el contexto de las metodologías ágiles de desarrollo para apoyar la práctica frecuente de liberaciones de código. El hecho que se practique la integración continua no significa que se esté aplicando DevOps (Lent, 2014).

Integración continua es uno más de los componentes que integran a DevOps. DevOps abarca muchos conceptos que es necesario tratarlos en este estudio para tener verdadero alineamiento Organización-DevOps-Usuarios que ayuden a cumplir los objetivos estratégicos corporativos.

Es necesario encontrar una ruta que permita alinear DevOps-Usuarios, y asegurar que los beneficios de aplicar las buenas prácticas de DevOps ayuden a superar la brecha entre operaciones y desarrollo.

La ruta de alineamiento necesita partir de la definición, identificación y desglose de todos los componentes que intervienen en el desarrollo y despliegue de aplicaciones.

2.1.2. Principios de DevOps

DevOps maneja principios que son parte de la estructura colaborativa y son utilizados en toda la etapa de desarrollo y despliegue de aplicaciones. Los principios en los que se desenvuelve DevOps son los siguientes (Hüttermann, 2012) (Humble & Farley, 2011):

- Integración Continua
- Entrega Continua
- Despliegue Continuo²

2.1.3. Integración Continua

La Integración Continua es la forma en la que el equipo de desarrollo de software integra su trabajo parcial o total, en un determinado tiempo establecido por el equipo de trabajo (Hüttermann, 2012).

Requiere de herramientas de automatización que son únicas para todo el equipo de desarrolladores. Estas herramientas ayudan a integrar en forma continua partes de código que son validados por pruebas automáticas, lo cual vuelve más eficiente el trabajo del equipo de desarrollo, ya que permite detectar fallos en etapas tempranas del ciclo de desarrollo.

La Integración continua de cada uno de los integrantes del equipo va a favorecer en los tiempos y calidad del producto.

² Según Martin Fowler, entrega continua y despliegue continuo son términos similares y crean cierta confusión en algunos autores. Él define la diferencia como una decisión de negocio sobre la frecuencia de despliegue en producción. Entrega continua se trata de mantener la aplicación en un estado en el que siempre es capaz de desplegarse en producción. Despliegue Continuo consiste en desplegar todos los cambios en producción, todos los días o con mayor frecuencia. (<http://martinfowler.com/delivery.html>)

La automatización de la integración continua ayuda a descubrir errores, debido a las continuas pruebas de unidad automatizadas (Hüttermann, 2012)(Humble & Farley, 2011).

La Integración Continua ayuda a establecer en forma ordenada los avances que pueda tener la iteración de las tareas que se están automatizando (Hüttermann, 2012)(Humble & Farley, 2011)(Guske & Kloss, 2009)(Rümmler & Schlag, 2014).

2.1.4. Entrega Continua

Una vez lograda la integración, se debe continuar con el ciclo para realizar el testing que, si es satisfactorio, permita que la aplicación sea desplegada.

En esta etapa se implementan todos los cambios en el código y se somete a un proceso de pruebas estandarizado con el fin de generar un producto que pueda ser pasado a producción.

La entrega continua requiere una aprobación manual. Los periodos de tiempo para lograr entregas continuas depende que la organización cuente con el equipo de desarrollo y operaciones adecuado. De esta manera se pueden presentar versiones candidatas en tiempos mínimos.

2.1.5. Despliegue Continuo

El despliegue continuo es una práctica que permite llevar los resultados de un proceso de desarrollo a un entorno similar al de producción donde las pruebas funcionales puedan darse a escala completa. El objetivo es detectar problemas en producción lo más rápido posible. Es el momento temprano en que el usuario interactúa con la aplicación, revisa sus requerimientos y se puede volver atrás en el desarrollo.

El Despliegue Continuo exige una configuración del ambiente de trabajo, que permita un funcionamiento efectivo de las versiones candidatas por parte de los usuarios. Se empieza con una pre-configuración durante todo el proceso de desarrollo y una configuración final antes que se termine la versión candidata.

La entrega continua requiere de una aprobación manual previo a su paso a producción; en el despliegue continuo el paso a producción se realiza de forma automática cuando se han satisfecho todos los criterios definidos.

El Despliegue Continuo visualiza tempranamente los fallos que puedan existir en la aplicación.

Realizando Despliegues Continuos, se logra entregar productos al usuario que pueden ser diarios o semanales, y así analizar fallos que van a ser detectados en forma temprana, debido al poco código que hay que revisar.

2.1.6. Antecedentes del uso de Herramientas y buenas prácticas como soporte a DevOps

El análisis de las herramientas DevOps fue abordado desde dos ámbitos, el primero, dar mayor efectividad al uso adecuado de los recursos de TI en beneficio de los objetivos de la organización, y el segundo desde el aspecto colaborativo entre desarrolladores y operadores.

Se consideró que la estructura de las herramientas debía estar basada en los siguientes requerimientos del negocio: disponibilidad, monitoreo, versionamiento de aplicaciones, y se incluyó a la integración continua que forma parte de los principios de DevOps.

Una vez determinada la clasificación de las herramientas DevOps, se procedió a implementarlas en el área de desarrollo y despliegue de aplicaciones de TI de la Empresa ESEE (en la sección 6.4.2.2 se expone más a detalle acerca la empresa ESEE).

Como resultado de la implementación destaca un breve análisis:

- Las herramientas DevOps fueron implementadas en las áreas de desarrollo y despliegue de aplicaciones.
- Se obtuvo resultados satisfactorios en el ámbito colaborativo. El Personal de las áreas de desarrollo y despliegue salió de su status quo, a trabajar en un nuevo ámbito colaborativo, con transmisión de conocimientos, entre otros, técnicos y procesos de la organización y propios de TI, por lo cual se disminuyeron los tiempos de codificación y entrega de aplicaciones.
- Sin embargo continuaron algunos inconvenientes que salían del ámbito de las herramientas DevOps. La automatización de tareas no siempre era la solución a los verdaderos problemas que tenía el usuario. La automatización seguía resolviendo requerimientos puntuales.
- Se pudo percibir que la automatización con apoyo de las herramientas DevOps se centró en el uso efectivo de ellas, mas no en el valor agregado que se podía obtener direccionándolas a objetivos corporativos.
- La automatización de tareas mostró un claro desentendimiento entre los objetivos de la organización y los requerimientos a ser automatizados.
- Al usuario solo le interesaba que sus tareas sean automatizadas y no conocía si existían objetivos estratégicos en la organización al cual se debía alinear.

Estos factores hicieron que se vea a DevOps, como un grupo más de las herramientas que eran utilizadas para el desarrollo y despliegue de aplicaciones.

Su uso, si hizo más efectivo el desarrollo y despliegue de las aplicaciones, pero no resolvió un alineamiento efectivo entre TI y la organización.

Para lograr un contexto de apoyo más amplio por parte de DevOps, había que trasladar el papel de tradicional de TI hacia una parte integral de la estrategia de la organización. Por ello esta premisa es muy acertada; para convertirse en una empresa exitosa, la estrategia de TI debe estar totalmente alineada con la estrategia del negocio (Henderson & Venkatraman, 1990).

De esa manera se empezó el análisis de varias metodologías y modelos de alineación estratégica como COBIT y SAM, a fin de encontrar un marco de referencia que ayude a efectivizar las herramientas DevOps en un ámbito de entrega de servicios estratégicos.

Para determinar cuál de las dos era la más adecuada se realizó el siguiente análisis a fin de justificar su uso.

2.2. Justificación de uso de SAM mediante análisis conceptual de procesos de COBIT

2.2.1. Introducción

Las organizaciones han involucrado a las Tecnologías de la Información y Comunicaciones (TIC) en su oferta de servicios, lo que hace que sean altamente dependientes de los recursos computacionales (Lopez Paz, Macia Péres, & Delgado Fernández, 2008). Esto obliga a las TIC a entregar prestaciones con altos niveles de disponibilidad para garantizar continuidad de los servicios.

Con el fin de establecer un marco de referencia básico del ambiente empresarial, se abordan los conceptos de gobierno empresarial, corporativo, negocio y TI.

El Gobierno Empresarial se define como “...el conjunto de responsabilidades y prácticas ejercidas por la junta directiva y la dirección ejecutiva con el objetivo de proporcionar dirección estratégica, asegurando que se cumplan los objetivos y verificando que los recursos de la organización se utilicen en forma responsable...”. (IFAC, 2004)(OECD, 2015) (Pérez, 2007).

El Gobierno Empresarial no solo se limita a la dirección empresarial interna y externa y, a la dirección y control de los interesados; sino también, se ha convertido en un tema en diversos ámbitos de negocio, en el cual se incluye a TI (Brand, Beens, Vuuregge, & Batenburg, 2011).

La Figura 2.1 representa una primera relación básica entre estos conceptos (IFAC, 2004)(Muñoz & Ulloa, 2011).

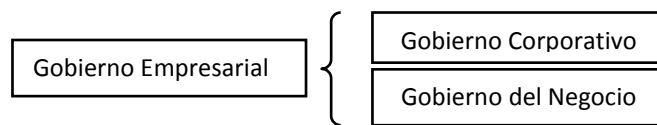


Figura 2.1 Estructura del Gobierno Empresarial

En la figura anterior se visualiza que la estructura del gobierno empresarial está compuesta por dos componentes: gobierno corporativo y de negocio.

El Gobierno Corporativo consiste en fijar estructuras organizacionales que establezcan los objetivos y lleven a cabo el seguimiento del desempeño de la organización para asegurar que dichos objetivos sean alcanzados (Céspedes, 2015).

El Gobierno de Negocio se encarga de la verificación y control del uso responsable de los recursos de la organización.

En esta estructura de gobierno empresarial, las TIC se presentan como un entorno de apoyo necesario para generar y sostener competencias (J. Fernández, Mayol, & Pastor, 2008).

Las competencias son generadas tomando como base la infraestructura de TI, mediante sus servicios de software y hardware, redes y comunicaciones, con el fin de alcanzar una comunicación constante entre los miembros internos y de otras organizaciones.

El alineamiento del negocio es necesario por la creciente dependencia de las TIC. Estas establecen la comunicación dentro y fuera de la organización, contribuyendo a las innovaciones de productos, o servicios.

El alineamiento del negocio requiere conocer el ámbito en el que se desenvuelve las TIC para alcanzar un soporte adecuado con sus objetivos.

Gobierno de TI trata acerca de la adecuada toma de decisiones sobre el uso de recursos de TI, que permita un aporte efectivo a los objetivos corporativos (Belalcázar et al., 2015)(Boar, 2001).

El Gobierno de TI comprende la especificación de los derechos de decisión y de rendición de cuentas para fomentar un comportamiento deseable en el uso de las TIC (Weill, Ross, No, & No, 2004) (Montaña, 2013).

El principal objetivo del gobierno de TI es facilitar y aumentar la habilidad de la organización para atender y cumplir con sus objetivos institucionales y para ofrecer la mejor información para la toma de decisiones relacionadas con la incorporación de TI a sus operaciones, programas y servicios a corto y largo plazo (Canay, 2014)(Céspedes, 2015)(Marulanda Echeverry, López Trujillo, & Cuesta Iglesias, 2009).

Estos conceptos son la base para definir un alineamiento estratégico entre los recursos de TI como factor preponderante en la generación del valor del negocio. Un nexo efectivo de alineamiento conlleva un beneficio mutuo entre el soporte estratégico de TI y la estrategia empresarial.

El alineamiento de TI y la organización debe medirse no solo por el grado en que TI apoya al negocio, sino también por la forma en que el negocio efectiviza las capacidades de TI.

La relación del Gobierno Corporativo y TI está marcada por la concordancia de algunos elementos: personas, infraestructura y procesos (A. Fernández & Llorens, 2009).

Para lograr esta adecuada relación entre el negocio y TI se revisó varios estándares de buenas prácticas como: COBIT, SAM. ISO/38500. De estos se tomó COBIT y SAM para establecer la guía más efectiva de alineamiento estratégico entre Negocio y TI.

2.2.2. Objetivo del análisis de relación entre COBIT y SAM

El objetivo del análisis de relación entre SAM y COBIT es verificar conceptualmente las interfaces que brinden el mejor alineamiento entre el negocio y TI.

Se analiza brevemente los conceptos relacionados con COBIT y el modelo SAM. La relación que se pretende encontrar entre ellos requiere de un análisis de procesos y dominios funcional. Se lo canaliza mediante la revisión de conceptos básicos de sus componentes, y no pretende ser un curso de buenas prácticas.

2.2.3. Análisis de procesos de COBIT

COBIT (Control Objectives for Information and related Technology), es el marco de referencia de buenas prácticas, que permite alinear los objetivos del negocio con el buen uso de los recursos de TI, a fin de alcanzar los objetivos del negocio.

COBIT Ayuda a habilitar los recursos de TI en beneficio de todo el negocio. Los principios y habilitadores se adaptan a todo tipo de empresas.

COBIT define sus procesos en base a la siguiente estructura (Ritegno, 2014)(Gary & Jimmy, 2008)(Tosar, 2013) (Molina & Rosales, 2010):

- Dominios, Procesos y Actividades
 - Evaluar, Dirigir y Monitorear (EDM)
 - Alineación, Planeación y Organización (APO)
 - Construir, Adquirir e Implementar (BAI)
 - Entregar, Servir y dar Soporte (DSS)
 - Monitorear, Evaluar y Valorar (MEA)

- Recursos de TI
 - Aplicaciones
 - Información
 - Infraestructura
 - Personas

- Requerimientos del Negocio
 - Efectividad
 - Eficiencia

- Confidencialidad
- Integridad
- Disponibilidad
- Cumplimiento
- Confiabilidad de la información

2.2.4. Principios de COBIT

Los 5 principios de COBIT son:

- I. Satisfacer las necesidades de las Partes Interesadas
- II. Cubrir la Compañía de Forma Integral
- III. Aplicar un solo Marco Integrado
- IV. Habilitar un Enfoque Holístico
- V. Separar el Gobierno de la Administración

I. Satisfacer las necesidades de las partes interesadas

La relación de las empresas con las diferentes partes interesadas requiere de un canal efectivo de comunicación mediante la creación de valor que genere beneficios mutuos. La oferta de servicios de software genera el camino adecuado para la entrega de prestaciones con valor agregado en beneficio de las partes interesadas.

Los servicios de software deben ser parte de la estrategia la Organización, alineados a satisfacer las necesidades de las diferentes partes interesadas.

II. Cubrir la empresa de extremo a extremo

COBIT no solo integra el gobierno corporativo y la gestión de TI sino la información relevante y los procesos de toda la empresa.

Se concentra en los procesos de gobierno y gestión de TI desde una perspectiva global. Se integra con las políticas empresariales mediante el buen uso de sus recursos. El desarrollo y despliegue de aplicaciones requiere la integración de funciones y procesos de gestión y gobierno que den fluidez a la entrega de aplicaciones efectivas.

III. Aplicar un marco de referencia único integrado

Un solo marco de referencia permite optimizar los recursos de la empresa. COBIT tiene una distribución de procesos de gobierno y gestión que son flexibles en la creación de diferentes interfaces con otros estándares, ofreciendo cada uno soporte en la interacción y control del uso adecuado de recursos de TI, a fin de estructurar objetivos empresariales más integrados.

IV. Hacer posible un enfoque holístico

En este principio se soporta el Gobierno de TI para visualizar en forma global la estructura corporativa. Lo realiza mediante catalizadores o habilitadores que ayuda a conformar un grupo sólido en la implementación del esquema de gobierno corporativo.

Se tiene los siguientes catalizadores:

- Principios, Políticas y Marcos de Trabajo
- Procesos
- Estructuras Organizativas
- Cultura, Ética y Comportamiento
- Información
- Servicios, Infraestructuras y Aplicaciones
- Personas, Habilidades y Competencias.

V. Separar el Gobierno de la Gestión

El marco de trabajo COBIT 5 permite diferenciar entre Gobierno y Gestión.

Gobierno es el responsable del establecimiento de estructuras organizacionales que establezcan los objetivos y lleven a cabo el seguimiento del desempeño de la organización para asegurar que dichos objetivos sean alcanzados (Céspedes, 2015).

La Gestión planifica, construye, ejecuta y monitorea las actividades conforme a las directivas fijadas por el ente de Gobierno para lograr los objetivos de la organización.

COBIT propone separar los procesos de gobierno que corresponden a Evaluar, Dirigir y Monitorear, de los procesos de Gestión: Planificar, Construir, Operar. Monitorear.

La separación es flexible ya que se puede adaptar a distintas necesidades de la organización. El objetivo es organizar los procesos para cubrir los mismos objetivos sin perder su capacidad de gobierno o gestión.

2.2.5. Procesos Habilitadores

COBIT divide las actividades y prácticas de la organización relacionadas con la TI en dos áreas principales: Gobierno y Administración:

- El Dominio de Gobierno contiene cinco procesos de gobierno; dentro de cada proceso se definen las prácticas para Evaluar, Dirigir y Monitorear (EDM).
- El Dominio de la Administración tiene cuatro dominios: Planificar, Construir, Operar y Monitorear.

2.2.6. Procesos de Gobierno

COBIT permite tener una estructura como guía para establecer el camino de alineamiento entre la directriz corporativa con la gestión de negocio.

Se analizan los conceptos que relacionan al marco de gobierno y sus los procesos EDM.

EDM01: Asegurar que se fija el marco de gobierno y su mantenimiento.

Establece y mantiene un marco de gobierno de acuerdo a su planificación estratégica.

Se enmarca en la evaluación de necesidades externas e internas y las convierte en servicios de acuerdo a su prioridad.

Establece el monitoreo como una función que permite realizar el seguimiento del desempeño a fin de cumplir los objetivos en su ámbito de conformidad.

Mantiene efectivos los procesos, roles y responsabilidades para brindar soporte a los objetivos empresariales.

EDM02: Asegurar la entrega de valor.

Permite asegurar la entrega de beneficios y servicios con valor agregado a las partes interesadas. Las estrategias externas e internas de la organización están encaminadas a organizar los recursos de infraestructura del negocio y TI, como procesos, actividades, acciones, para generar prestaciones que brinden valor agregado a todos los interesados.

EDM03: Asegurar la optimización de riesgos.

Asegura que los recursos adquiridos por TI brinden prestaciones con beneficios directos a la organización, al cliente externo y al usuario interno, en el marco de optimización de recursos administrativos y de TI.

Asegura que los riesgos de la empresa relacionados con TI no superen la tolerancia al riesgo.

El buen uso de los recursos de la organización mediante la sincronización y optimización de procesos, personal y actividades, y componentes de TI, se verán reflejados en la disminución de riesgo en la entrega de prestaciones al usuario.

EDM04: Asegurar la optimización de recursos.

Asegura que los recursos de la organización sean manejados con eficiencia, obteniendo los mayores beneficios con los mínimos costes.

La eficiencia tiene una estrecha relación con la eficacia (ya que esta hace énfasis en los resultados, lograr objetivos, crear valores) para optimizar recursos no tendría que ser solo eficiente sino que también ser eficaz.

EDM05: Asegurar la transparencia hacia las partes interesadas.

Asegura que los intereses de las partes interesadas sean canalizados en forma eficiente. Cuando los accionistas y otros interesados tienen expectativas en una organización la cual está guiada bajo normas y principios que regulan el diseño, integración y funcionamiento de los componentes de gobierno de la empresa, estas expectativas deben ser canalizadas mediante requerimientos que ayuden a satisfacer los objetivos de lado y lado, tanto accionistas como partes interesadas.

Este vínculo requerimiento-satisfacción se identifica con la representación de la relación cliente - servidor. En cadena hacia abajo desde los accionistas y otros interesados, quien con sus requerimientos se asumen como clientes, hasta los niveles operativos pasan a ser clientes y servidores dependiendo del requerimiento recibido y de la entrega del servicio para satisfacer los requerimientos solicitados.

2.2.7. Procesos de Gestión

APO01: Administrar el marco de administración de TI

La administración de TI se encuentra definida por la planificación estratégica de TI. La implantación de mecanismos para alinear los recursos de TI con los objetivos, metas, planes y proyectos, acorde a las políticas y principios corporativos.

APO02: Administrar la estrategia

Las estrategias externas e internas de TI proporcionan una guía hacia donde se deben dirigir los esfuerzos de todo el componente empresarial. Aprovechar los recursos de la organización y TI, para obtener respuestas ágiles de acuerdo a los objetivos estratégicos definidos.

APO03: Administrar la arquitectura corporativa

Establecer la arquitectura corporativa enmarcada en: normas, políticas, directrices soporte de la infraestructura de la organización: procesos administrativos de negocio, personas, infraestructura de TI: procesos relacionados con los servicios de TI, capas de arquitectura tecnológica como componentes de hardware, software, con el objetivo de

mejorar los atributos de calidad, eficiencia, eficacia con los mínimos costos, es la misión de toda organización.

APO04: Administrar la innovación

La innovación nace del cruce de ideas que impulsan la creación de nuevos productos dentro de la organización. Los recursos con los que cuenta y los objetivos empresariales hacia dónde quiere llegar, brindan el marco apropiado para identificar las propuestas de mejoras administrativas y tecnológicas que influyan en obtener beneficios con valor agregado y brinden una mejor toma de decisiones.

APO05: Administrar el portafolio

La planificación estratégica empresarial delinea ejes estratégicos, objetivos, metas, planes, proyecto y acciones estratégicas, que deben ser canalizadas de acuerdo a los más altos intereses de la organización (Kaplan & Norton, 2004).

APO06: Administrar el presupuesto y los costos

La planificación estratégica define un portafolio de proyectos a ser ejecutados. Cada uno de ellos lleva su costo relacionado. El seguimiento y control adecuado en el cumplimiento de cada etapa que ayude a identificar fallos y aplicar los correctivos adecuados, va a permitir un adecuado uso de recursos encaminados a obtener metas específicas a bajo costo.

APO07: Administrar el recurso humano

La planificación de los roles, responsabilidades, a fin de garantizar que las capacidades, conocimientos y habilidades del personal que lleva adelante los proyecto o tareas rutinarias, proporcionen tareas y actividades que cumplan las expectativas de desempeño.

APO08: Administrar las relaciones

Gestionar las relaciones entre la organización, TI deben estar enmarcadas con un enfoque colaborativo a fin de establecer planificaciones conjuntas cuyo objetivo es lograr resultados en el marco de confianza y respeto mutuo.

APO09: Administrar los contratos de servicio

Los acuerdos de niveles de servicio se coordinan con la planificación de requerimientos y servicios solicitados y entregados, a fin de determinar con exactitud las prestaciones que serán entregadas al usuario. La disponibilidad de recursos debe estar claramente definida con sus fortalezas y limitantes, a fin de encontrar los rangos en los cuales se entrega el servicio.

APO10: Administrar los proveedores

La gestión de proveedores conlleva varias aristas: provisión de servicios de TI por proveedores externos, tareas de planificación a fin de identificar la capacidad que tiene un proveedor en relación a los recursos utilizados en la organización. El aporte de un proveedor se lo mide en base a la capacidad de respuesta al ser solicitado un componente que es necesario para una actividad clave.

APO11: Administrar la calidad

La identificación y definición de modelos, métodos, indicadores para evaluar procesos, planes proyectos y actividades, van a permitir estándares de mejora.

APO12: Administrar los riesgos

La planificación de riesgos para identificar y evaluar los componentes de la organización que sean susceptibles de errores, fallos, averías e impidan el normal desenvolvimiento de su tarea, debe llevar acciones que permitan el control preventivo, detectivo y correctivo de un servicio.

APO13: Administrar la seguridad

Abarca la gestión de planificación de seguridad en todos los ámbitos y componentes de TI.

MEA01: Monitorear, evaluar y valorar el desempeño y cumplimiento

Este proceso se encarga de la identificación de métricas empresariales que ayuden a supervisar y evaluar los procesos en el marco de conformidad.

MEA02: Monitorear, evaluar y valorar el Sistema de Control Interno

Este proceso se encarga de la identificación de métricas que ayuden a supervisar y evaluar los procesos en el marco del desempeño.

MEA03: Monitorear, evaluar y valorar el cumplimiento con requisitos externos

Este se encarga de supervisar y evaluar el cumplimiento de requisitos de la organización relacionados con TI.

BAI01: Administrar programas y proyectos

Se encarga de planificar y gestionar el portafolio de los programas y proyectos, con el fin de encaminarlos en base a los objetivos de corto, mediano y largo plazo.

BAI02: Administrar la definición de requerimientos

Este proceso se encarga de identificar los requerimientos solicitados a fin de determinar los recursos de negocio o TI necesarios para dar soluciones viables o adecuadas.

BAI03: Administrar la identificación y construcción de soluciones

Se encarga de planificar el diseño, desarrollo, pruebas, y post-implementación, de componentes administrativos o computacionales a fin de alinearlos con los objetivos de la organización.

BAI04: Administrar la disponibilidad y capacidad

Se encarga de determinar los mecanismos que lleven a equilibrar los recursos y servicios con niveles de disponibilidad fiables.

También permite la evaluación de las capacidades de cada componente que interviene en la provisión de servicios a fin de prever posibles impactos con la falta de recursos.

BAI05: Administrar la habilitación del cambio

Se encarga de gestionar la implementación del cambio relacionado con procesos de la organización y TI.

BAI06: Administrar los cambios

Se encarga de gestionar el ciclo de vida de los cambios en ámbitos relacionados con procesos corporativos y TI. Incluye normas, políticas, procedimientos.

BAI07: Administrar la aceptación de cambios y transiciones

Se encarga de gestionar que la solución propuesta se encuentra en base a los requerimientos solicitados. Administra la etapa de implementación y producción, en base a evaluaciones periódicas del producto final.

BAI08: Administrar el conocimiento

Se encarga de planificar que el conocimiento del personal necesario para desarrollar soluciones sea el óptimo a fin de que la solución se haga efectiva en el menor tiempo. Gestiona la capacitación adecuada a fin de alcanzar los objetivos deseados.

BAI09: Administrar los activos

Se encarga de administrar los recursos tangibles e intangibles a fin de asegurar que su uso aporte con valor agregado a las necesidades para las cuales fueron adquiridos.

BAI10: Administrar la configuración

Se encarga de gestionar la configuración adecuada de los componentes necesarios para dar soluciones efectivas a la prestación de servicios.

DSS01: Administrar las operaciones

Se encarga de coordinar, monitorear y ejecutar los recursos relacionados con la administración de sistemas.

DSS02: Administrar las solicitudes de servicios y los incidentes

Se encarga de planificar y coordinar la identificación, clasificación, codificación de incidentes de acuerdo a procedimientos y actividades, para proveer una respuesta oportuna en el ámbito de primer nivel o al ser escalado a personal especializado.

DSS03: Administrar los problemas

Se encarga de planificar y coordinar la identificación, clasificación, codificación de problemas de acuerdo a estándares, procedimientos y actividades, a fin de prevenir los incidentes recurrentes.

DSS04: Administrar la continuidad

Establecer una planificación que permita definir los recursos por cada servicio entregado por los componentes de TI, a fin de lograr un equilibrio entre disponibilidad y costos.

DSS05: Administrar los servicios de seguridad

Gestionar la identificación de información, datos y servicios estratégicos a fin de protegerlos de acuerdo a políticas establecidas para el efecto.

DSS06: Administrar los controles en los procesos de negocio

Gestiona la identificación de controles apropiados y relacionados con procesos del negocio. Asegurar que los procesos y recursos de TI tengan relación con procesos optimizados de la organización a fin de lograr una organización adecuada de recursos.

2.2.8. Modelo de Alineamiento Estratégico (Strategic Alignment Model. SAM)

El Modelo de Alineamiento Estratégico (SAM) permite estructurar la estrategia global de una organización mediante cuatro dominios (Henderson & Venkatraman, 1990)(Vargas, Boza, & Cuenca, 2011).

- Estrategia del Negocio
- Estrategias de TI
- Procesos e Infraestructura Organizacional
- Procesos e Infraestructura

SAM incluye estructuras organizacionales llamadas dominios, cada una con sus propios procesos, habilidades o capacidades operativas.

Se establece un modelo organizacional mediante dominios que habilita la relación entre las estrategias y recursos de negocio y TI (Ver Figura 2.2).

Las diferentes relaciones de SAM permiten visualizar las influencias ejercidas entre cada uno de los dominios, sin dejar de lado a ninguno de ellos ya que son transversales en su afectación.

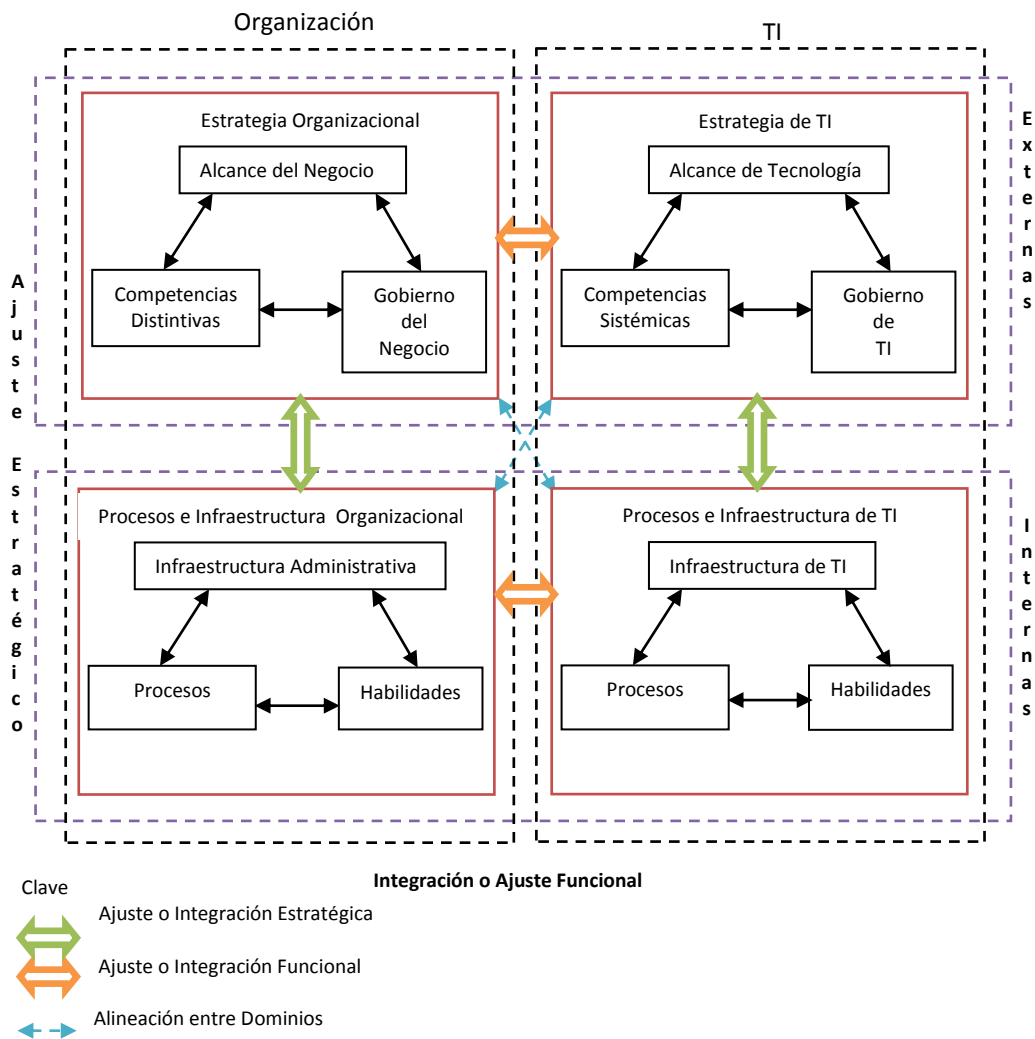


Figura 2.2 Modelo de Alineación Estratégica (Henderson & Venkatraman, 1990)

Las flechas del gráfico anterior tienen distintos colores con el fin de visualizar más claramente los diferentes ajustes e influencias ejercidas entre cada uno de los dominios. El ajuste o Integración Estratégica enlaza dos dominios, verticalmente y horizontalmente, también es llamado ajuste bivariante, que es el tipo más simple de relación (Henderson & Venkatraman, 1990).

La relación bivariada verticalmente (en la Figura 2.2 identificada con flecha verde) tiene dos aristas:

- Entre la estrategia de negocio y la infraestructura y los procesos de la organización.
 - Correspondientemente, se especifica una relación bivariada entre la estrategia de TI y los procesos e infraestructura de TI.
- Destacando la necesidad de interconectar el posicionamiento externo de una organización en el mercado con su enfoque de gestión de la función externa de TI en su contexto organizacional.

El ajuste o integración funcional, enlaza horizontalmente, las otras dos relaciones bivariadas (en la Figura 2.2. identificada con flecha naranja):

- Los vínculos entre la estrategia empresarial y la estrategia de TI (es decir, la articulación del alcance de TI requerido, el desarrollo de competencias sistémicas y los mecanismos de gobernanza de TI) refleja la capacidad de aprovechar la estrategia de TI para dar forma y apoyo a la estrategia empresarial.
- De manera correspondiente, el vínculo entre la infraestructura y los procesos de la organización y la infraestructura y los procesos de TI, refleja la necesidad de asegurar una coherencia interna entre los requisitos y expectativas organizacionales, por un lado, y la capacidad de entrega dentro de la infraestructura de TI.

La Alineación entre dominios, está representada en la Figura 2.2 por las flechas celestes, está representada por dos tipos de alineación diagonal:

- La alineación entre dominios entre la estrategia de negocio y los procesos e infraestructura de TI, representa una visión tradicional de trabajo propia de paradigmas empresariales con la que se viene trabajando hasta la actualidad. El negocio solicita recursos de TI, y este debe reflejar el potencial de tecnología para cambiar los procesos organizativos y TI en beneficio de las estrategias empresariales.
- El otro tipo de alineamiento, entre la estrategia de TI y la infraestructura y los procesos organizativos, refleja que la estrategia empresarial interna debe descomponerse en procesos de trabajo para definir los requisitos de la infraestructura y los procesos de TI de acuerdo de su estrategia externa.

2.2.9. Alcance del modelo SAM

A continuación se detallan los alcances de cada uno de los dominios (Henderson & Venkatraman, 1990)(Jiménez, 2011).

- En el Dominio de la Estrategia del Negocio, las directrices ejecutivas están encaminadas a definir estrategias que permitan tener alineamientos entre los distintos dominios para cumplir los objetivos empresariales planteados. El Dominio de la Estrategia Negocio se distingue por las características de los siguientes componentes:
 - Alcance del Negocio
 - Competencias Distintivas
 - Gobierno de Negocio

El alcance del negocio es la selección de ofertas de productos que se ofrecerá al mercado. Puede ser una empresa de provisión de energía, o de servicios tecnológicos o de servicios educativos. Puede ser público o privado.

El Alcance del Negocio es el eje que permite visualizar el entorno del usuario/cliente al cual va dirigido el producto. Analiza la dimensión del servicio o producto ofertado con el fin de establecer estrategias de acuerdo al tipo de usuario que va a utilizar su producto y que le permita obtener valor agregado de

la competencia. Define las actividades claves de la organización para llegar a un potencial usuario de su producto.

Las Competencias Distintivas son los atributos de una organización que le distinguen y le dan valor agregado y ventaja competitiva sobre la competencia.

El Gobierno del Negocio se distingue por el manejo adecuado de su infraestructura que incluye actividades y recursos de la organización a fin de dar fluidez y efectividad a la entrega de productos y servicios ofertados en el ámbito interno y externo.

Las alianzas estratégicas que ayuden a estructurar operaciones y relaciones para entregar en forma adecuada los productos o servicios al usuario, son parte de las actividades que debe enfrentar el Gobierno del Negocio.

- El Dominio de Procesos e Infraestructura Organizacional define la estructura de organización y control más adecuado a fin de establecer roles, responsabilidades, procesos, actividades para coordinar la entrega adecuada de recursos del negocio.

El Dominio de Procesos e Infraestructura Organizacional está constituido por:

- La Infraestructura Administrativa
- Los Procesos
- Las Habilidades

Las capacidades de la Infraestructura Administrativa, deben estar acorde con las características del tipo de servicio que se pretende entregar al usuario.

Los procesos deben mantener una articulación entre flujo de trabajo e información optimizados a fin de llevar a cabo las actividades claves con factores de eficiencia.

Las habilidades definen las capacidades de los individuos y la organización para ejecutar las tareas clave que apoyan los objetivos estratégicos del negocio.

- El Dominio de Estrategia de TI consiste en definir una planificación estratégica de TI que apoye con sus componentes computacionales a la modernización empresarial.

El Dominio de Estrategia de TI tiene tres componentes:

- Alcance de TI
- Competencias Sistémicas
- Gobernabilidad de TI

El Alcance de TI es el conjunto de componentes que tiene TI para dar soporte a los requerimientos corporativos que den valor agregado a las tareas del usuario.

Las Competencias sistémicas se enfocan en los atributos distintivos de las competencias de TI: Mayor fiabilidad del sistema, escalabilidad, interconectividad, que contribuyen a la creación de nuevas estrategias del negocio.

La Gobernabilidad de TI son los mecanismos que utiliza TI para aplicar el uso de los componentes computacionales acorde a las necesidades de los objetivos corporativos.

- El Dominio de Procesos e Infraestructura de TI consiste en alinear los procesos y componentes computacionales en la modernización corporativa.
El Dominio de Procesos e Infraestructura de TI, está representado por tres dimensiones:
 - La Infraestructura de TI
 - Los Procesos de TI
 - Las habilidades

La Infraestructura de TI es la administración de los recursos computacionales, de software, hardware y administración adecuada para ofertar servicios con disponibilidad adecuada al usuario.

Los Procesos de TI, consisten en la optimización de los procesos propios de TI para lograr una sincronización adecuada de las áreas de TI en la oferta de servicios al usuario.

Las Habilidades, determinan las capacidades de todo el personal técnico, operativo, administrativos para lograr alineamientos efectivos en los trabajos operativos necesarios para la entrega de prestaciones adecuadas al usuario.

Las relaciones que se establecen entre los cuatro dominios determinan la dependencia mutua de las estrategias y los procesos generales de la organización.

Estas relaciones permiten visualizar las influencias ubicuas internas y externas entre los dominios de Gobierno de Negocio, TI y otros interesados.

SAM plantea competencias distintivas tanto de procesos como de sistemas en su modelo. No plantea el uso de herramientas en su estructura.

Efectiva utilización de TI requiere del alineamiento de sus estrategias con las de la organización. Y plantea dos dimensiones de estrategias de alineamiento (Henderson & Venkatraman, 1990)(Avison, Jones, Powell, & Wilson, 2004)(Ojo, Janowski, Shareef, & Estevez, 2009):

- Alineamiento TI-Organización
 - Las Estrategias de TI son desarrolladas para dar un claro soporte a las Estrategias Organizacionales
- Alineamiento Organización-TI
 - La Estrategia Organizacional es desarrollada para aprovechar oportunidades desarrolladas por las Estrategias de TI

Los requerimientos de los Accionistas y Otros Interesados se van transformando conforme cambian los Dominios. El requerimiento superior, es decir realizado por dominios gobernantes se toma como obligatorio. Para ser cumplido debe ser analizado por la Estrategia que le corresponde de acuerdo a su competencia.

Para dar soporte a las Estrategias de la Organización y TI, la Infraestructura del Negocio y TI deben estar alineadas de acuerdo a la estrategia que adopten los ejecutivos de la organización para hacer frente a los objetivos corporativos planteados.

SAM considera a la estrategia de negocio y la de TI como factores externos. Y los dominios de infraestructura y procesos, como factores internos. La relación entre la perspectiva externa (estrategia de TI), e interna (infraestructura y procesos de TI), determina los alcances que puede llegar a tener TI dentro de la organización como soporte a los objetivos de la organización.

Esta distinción implica dos niveles de integración entre componentes de la estrategia y la estructura de la organización (Avison et al., 2004)(Henderson & Venkatraman, 1990):

- *Ajuste o Integración Estratégico externo del negocio y estrategia externa de TI*, que establece el potencial y capacidad de soporte que tiene TI a nivel estratégico.

El dominio externo se refiere a la posición de una organización relativa a sus competidores, relación producto-mercado, alianzas estratégicas y otros interesados.

Ajuste o Integración operativa o funcional, que es el vínculo entre la infraestructura de TI y los procesos y la infraestructura interna de la organización y los procesos.

El dominio interno se refiere a la estructura de la organización, procesos, habilidades.

Estos ajustes integran las perspectivas de los dominios de SAM expuestos en la Figura 2.2, bajo el razonamiento que los ajustes estratégicos y funcionales por sí solos no pueden realizar un alineamiento eficaz. Ese el caso de los ajustes bivariantes vistos en párrafos anteriores.

En base a este argumento se presenta las distintas perspectivas de co-alineación o alineación multi-variable entre las estrategias funcionales y estratégicas.

El vínculo entre la estrategia, la infraestructura y los procesos se examina en términos de proceso, estructura y personas, y no en un nivel abstracto de intentar relacionar las arquitecturas internas con los objetivos estratégicos (Avison et al., 2004).

Las perspectivas de co-alineación o alineamiento múltiple o de varios dominios, trabajan sobre la premisa de que la alineación estratégica en un nivel organizacional sólo puede ocurrir cuando tres de los cuatro dominios corporativos intervienen en el proceso de alineamiento. La premisa central es que el cambio no puede ocurrir en un dominio sin afectar en al menos dos de los tres dominios restantes de alguna manera.

La perspectiva de alineación de una organización puede derivarse dibujando una línea a través de los tres dominios intervinientes: dominio de anclaje, dominio de pivote y dominio impactado (Henderson & Venkatraman, 1990).

El dominio de anclaje es el dominio que inicia el cambio o modernización de la organización. Puede tener la representación más fuerte en el nivel ejecutivo o ser el área estratégica del negocio que encabece la modernización del negocio.

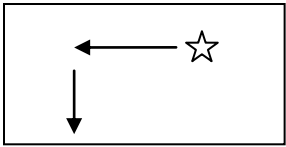
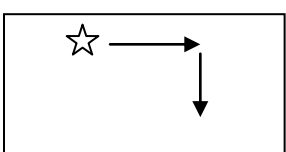
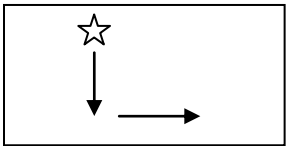
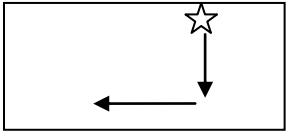
El dominio de pivote indica qué dominio funcional o estratégico será finalmente afectado por el cambio iniciado dentro del dominio de anclaje.

Dominio Impactado es el más afectado por el cambio iniciado en el Dominio de anclaje, ya que debe modificar su estrategia para dar soluciones y viabilidad a las acciones solicitadas por el dominio de anclaje y pivote.

La dirección de la perspectiva se extiende desde el dominio de anclaje al dominio impactado, a través del dominio de pivote. Las perspectivas son de "arriba hacia abajo" o impulsadas por el proceso de "abajo hacia arriba" (Henderson & Venkatraman, 1990).

La alineación entre dominios se la realiza entre tres dominios vinculados secuencialmente. Si bien se puede realizar ocho combinaciones (Ver Figura 2.2), cuatro son particularmente importantes y de gestión relevante y se etiquetan como: implementación de la estrategia, explotación de la tecnología, apalancamiento tecnológico e implementación de la tecnología (Ver Tabla 2-1).

Tabla 2-1 Cuatro perspectivas dominantes sobre la planificación de TI³

Etiqueta	Perspectiva de alineamiento entre dominios	Anclaje entre dominios
(1) Potencial Competitivo		Estrategia de Tecnología
(2) Transformación de la tecnología		Estrategia del Negocio
(3) Estrategia de Ejecución		Estrategia del Negocio
(4) Nivel de Servicio		Estrategia de Tecnología

(1) El Potencial Competitivo, se refiere al uso adecuado de las capacidades de TI para impactar en las dimensiones claves de la estrategia empresarial, como el apoyo en la creación de nuevos productos y servicios (es decir, el alcance del negocio), influir en los atributos clave de la estrategia (competencias distintivas) y desarrollar nuevas formas de relaciones (es decir, la gobernanza empresarial).

En la perspectiva del Potencial Competitivo, TI habilita oportunidades estratégicas. La Estrategia de TI propone la modernización de la organización en base a las competencias distintivas de cada área de TI. La Estrategia del Negocio brinda su potencial apoyo a la Estrategia de TI. La Infraestructura del Negocio apoya a las Estrategias de TI y del Negocio de acuerdo a los lineamientos de cambios que se presenten.

(2) La Transformación de Tecnología, como se muestra en la Tabla 2-1, es una perspectiva de alineamiento de varios dominios, que evalúa las implicaciones de la implementación de la estrategia de negocio elegida a través de una adecuada estrategia de TI y la articulación de la infraestructura funcional y procesos sistémicos requeridos. La razón potencial es que la estrategia empresarial se ejecuta con mejor alcance mediante el aprovechamiento de las capacidades tecnológicas más que el aprovechamiento del diseño de una organización interna eficiente.

³ Tomado de (Henderson & Venkatraman, 1990)

En la perspectiva de la Transformación de Tecnología, la Estrategia del Negocio propone la modernización con apoyo directo de la Estrategia e Infraestructura de TI. La Estrategia de TI se adapta a las directrices de modernización organizacional. La Infraestructura de TI apoya con todos los servicios que así lo requiera la Estrategia de TI.

(3) La Estrategia de Ejecución, como se muestra en la Tabla 2-1, establece la evaluación de las implicaciones de la implementación de la estrategia de negocio elegida a través de la infraestructura y procesos de gestión de la organización, así como el diseño y desarrollo de la infraestructura interna de TI requerida.

En la perspectiva de la Estrategia de Ejecución, el Dominio del Negocio es el que propone los cambios en la organización con apoyo de la Infraestructura del Negocio y la Infraestructura de TI. Los ambientes de las áreas de Infraestructuras de Negocio y TI sirven de apoyo y pueden adaptar sus procesos de acuerdo a la estrategia de cambio que solicite la Estrategia de Negocio.

(4) El Nivel de Servicio, como se muestra en la Tabla 2-1, se refiere a la adecuación estratégica, entre la articulación externa de la estrategia TI y la implementación interna de la infraestructura y procesos de TI con el correspondiente impacto en la infraestructura y los procesos organizacionales. En esta perspectiva, el papel de la estrategia empresarial mínimo e indirecto, sin embargo su influencia es fuerte, debido al apoyo administrativo que puede brindar para la organización interna.

La Estrategia de TI propone una adecuada provisión de servicios de TI siendo parte de esta la información que es considerada como un producto vital. El enfoque es habilitar la infraestructura de TI mediante el ajuste de la infraestructura de negocio servicio de la estrategia de TI.

2.2.10. COBIT vs SAM

En el siguiente análisis se presentan varias consideraciones que diferencian al modelo SAM de COBIT:

- COBIT tiene identificados procesos de gobierno y de gestión: EDM, APO, BAI, DSS, MEA, esta es su gran fortaleza.
- SAM no tiene ese nivel de abstracción, queda a criterio de la persona que va a implementar el modelo, la abstracción de componentes en cada dominio.
- COBIT no tiene identificado un flujo de integración entre las distintas capas de procesos.
- SAM maneja perspectivas de alineamiento entre sus dominios, con la premisa que los ajustes estratégicos y funcionales por sí solos no pueden realizar un alineamiento eficaz. Este ajuste o integración permite alinear la estrategia externa e interna del negocio y TI, y los abstrae mediante la derivación a través de los tres dominios dominantes: dominio de anclaje, dominio de pivote y dominio impactado.

2.2.11. Definición de SAM como modelo de alineamiento estratégico

Para definir la relación entre SAM y COBIT se identifican los siguientes elementos:

En COBIT se determinan los procesos relacionados con la relación externa e interna empresarial y TI.

En SAM se utiliza el ajuste estratégico como referencia.

Los procesos de COBIT que están íntimamente ligados al tema de gobierno, podríamos seleccionar los siguientes:

- EDM01 Asegurar que se fija el marco de gobierno y su mantenimiento
- EDM02 Asegurar la entrega de valor
- EDM03 Asegurar la optimización de los riesgos
- EDM04 Asegurar la optimización de los recursos
- EDM05 Asegurar la transparencia de las partes interesadas
- MEA02 Monitorear, Evaluar y Valorar el Sistema de Control Interno
- MEA03 Monitorear, Evaluar y Valorar el Cumplimiento con Requisitos Externos
- APO01 Administrar el marco de la administración de TI
- APO11 Administrar la calidad

Los procesos de COBIT ligados al tema de gestión son los siguientes:

Monitorear, evaluar y valorar:

MEA01

Alinear, Planear y Organizar

APO02, APO03, APO04, APO05, APO06, APO07, APO12, APO13.

Construir, adquirir e implementar

BAI01, BAI02, BAI03, BAI04, BAI05, BAI0106, BAI0107, BAI0108, BAI09, BAI10.

Entregar, servir y dar soporte

DSS01, DSS02, DSS03, DSS04, DSS05, DSS06.

En base a esta identificación, en la Figura 2.3 se presentan las interfaces y alcances de SAM y COBIT.

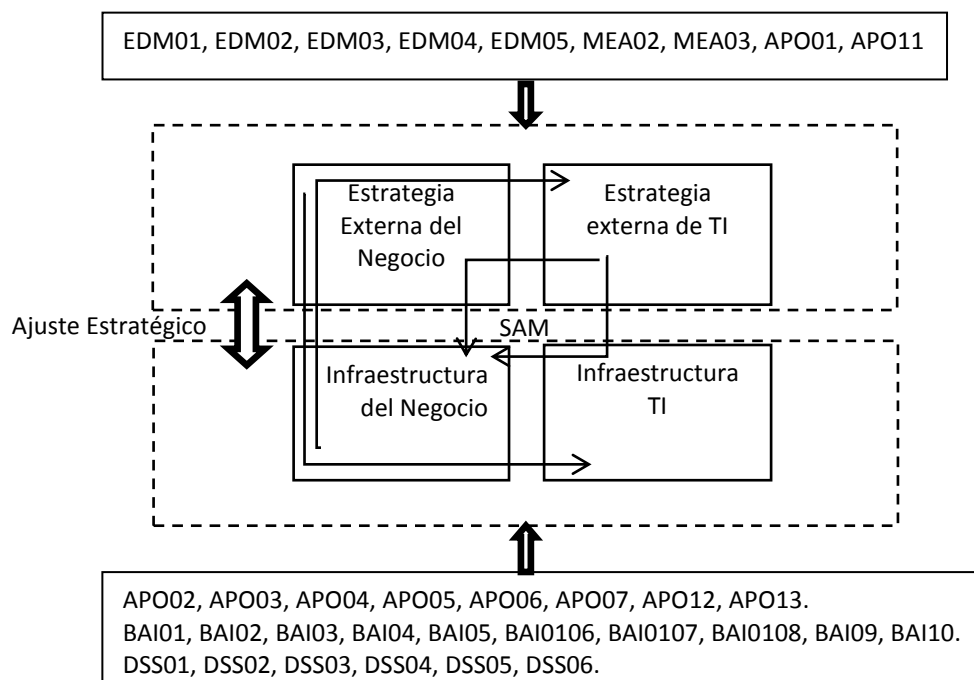


Figura 2.3 Interfaz entre Dominios de SAM con Procesos de COBIT

La explicación de la interfaz mostrada en la figura anterior es la siguiente:

- En la figura anterior se abstraen procesos no áreas.
- SAM maneja dos niveles de integración
 - o el ajuste estratégico externo del negocio y la estrategia externa de TI
 - o el ajuste funcional que es el vínculo entre la infraestructura de TI y los procesos y la infraestructura interna de la organización y los procesos
- Es posible incluir los procesos de gobierno y gestión de COBIT enmarcándolos en los ajustes estratégicos y funcionales.
- Las distintas perspectivas de co-alineación o alineación multi-variable entre los ajustes funcionales y estratégicos, con el apoyo de los procesos de COBIT, definen una estructura muy robusta en el alineamiento estratégico corporativo y TI.
- Se puede realizar la abstracción de los procesos de COBIT en las perspectivas de SAM de acuerdo a los ajustes estratégicos o funcionales que se desee obtener.
- Los Dominios de Anclaje, de Pivote y de Impacto, con la interfaz colaborativa de los procesos de COBIT, pueden alcanzar diversas perspectivas impulsadas de arriba hacia abajo o de abajo hacia arriba.
- El modelo SAM permite establecer ocho combinaciones entre dominios, que con el aporte de los procesos de COBIT, se logra una gran fortaleza en la búsqueda de la mejor alternativa de alineamiento corporativo y TI.

2.2.12. Conclusiones

- Se puede determinar que SAM y COBIT pueden trabajar juntos. COBIT entrega procesos que SAM no los tiene definidos.
- SAM tiene definidos anclajes de dominio que no los tiene COBIT.
- Los dominios de SAM y procesos de COBIT no se traslapan. Se colaboran que es el principio de DevOps.
- En el modelo SAM los procesos y recursos de COBIT se pueden ir derivando de acuerdo a la necesidad de la actividad a realizar.
- Como conclusión determinante para el uso de SAM como modelo de alineación estratégico es: la relación de dominios de SAM lleva una connotación más fuerte para el trabajo colaborativo de DevOps. Ayuda a determinar con más efectividad los requerimientos estratégicos.

Al definir a SAM como modelo de alineamiento estratégico, también se determinó que era necesario apoyarlo con los principios de la metodología MLS y con las buenas prácticas de ITIL. (Ver Figura 2.4).

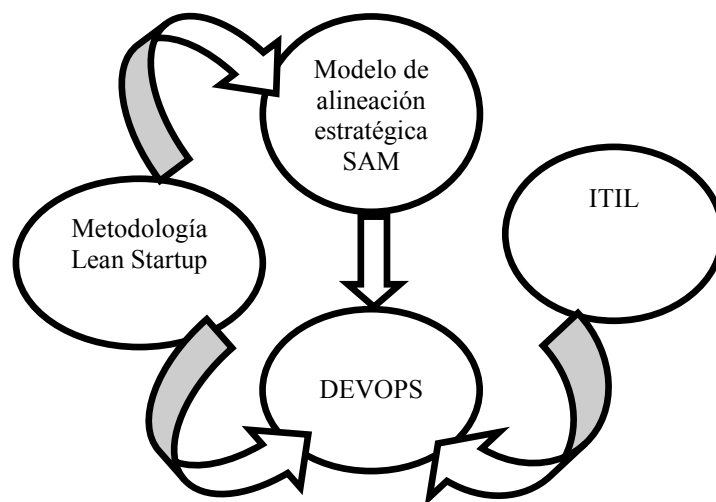


Figura 2.4 Propuesta de modelos, y metodologías, y buenas prácticas a ser estudiadas como soporte a DevOps

En la Figura 2.4 se identifica a DevOps como eje central que tiene el soporte del modelo de alineamiento estratégico SAM, de la metodología MLS y tiene influencia de las buenas prácticas ITIL.

2.3. Metodología Lean Startup (MLS)

MLS se utiliza en esta tesis como guía de metodología de buenas prácticas, cuyo apoyo al alineamiento entre gobierno y TI, se efectiviza mediante la aplicación de sus principios en dos procesos:

- En el proceso de búsqueda de requerimientos que entreguen servicios con valor agregado al usuario, al aplicar SAM-RS.
- En el desarrollo y despliegue de aplicaciones con DevOps.

La característica de la MLS es plantear un ambiente libre que ayude al fluir de las ideas e innovación en las propuestas de transformación en las organizaciones (Ries, 2011)(Gaffney et al., 2014).

La metodología Lean Startup es un enfoque basado en principios para el desarrollo de nuevos productos (Flores, 2013).

Un Startup es una institución humana diseñada para crear un nuevo producto o servicio bajo condiciones de incertidumbre extrema.

Los principios en los que basa la metodología Lean Startup son (Ver Figura 2.5):

- Proponer Ideas
- Crear
- Medir
- Aprender
- Optimizar
- Pivotar

El elemento proponer ideas de la metodología Lean Startup define los aportes que entrega el personal relacionado con el ambiente de la organización. Las Ideas tienen una relación cercana con la Innovación.

El principio básico de la Innovación es el aporte de nuevos servicios o productos que den valor agregado a los clientes y por ende a la organización.

La Innovación y la Idea por sí solas no representan un valor agregado como tal.

La transformación de una Idea que se convierte en producto o servicio representa la Innovación (Crom, 2005).

El elemento Crear permite generar un producto mínimo viable y desplegarlo para uso del usuario en el menor tiempo posible.

El elemento Crear en este estudio se lo manifiesta desde dos ópticas:

- Producir componentes que pueden ser productos o servicios desde el punto de vista corporativo
- Producir componentes desde el ambiente de apoyo a la organización de acuerdo a los objetivos corporativos

La Métrica se define como la medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo.

Los atributos de las métricas se dan por acuerdo de un protocolo entre directivos y personal operativo, en base a un valor que mide la gestión de un área en un tiempo determinado.

El elemento Aprender permite abstraer los requerimientos y necesidades reales del usuario de acuerdo a los servicios que necesita.

Al tener ciclos cortos de generación de productos mínimos viables se aprende a revisar en tiempos de ciclos cortos las verdaderas necesidades del usuario.

En caso que el producto sea viable se continúa con la optimización.

El elemento Optimizar permite organizar en forma adecuada los procesos relacionados con los servicios que presta la organización a los usuarios.

El planteamiento de optimización debe ser coherente entre los servicios a los clientes externos e internos con el fin de lograr apoyos directos tanto de ejecutivos como personal de apoyo para obtener beneficios loables en la entrega de prestaciones.

En caso que el producto no sea viable, tenga o no la factibilidad de ser exitoso, la opción es realizar un Pivotaje que permita cambiar la idea original a ideas innovadoras que den valor agregado al producto (Blank & Dorf, 2012).

El Pivotaje se presenta desde 2 ópticas:

- f1: Pivotaje desde el punto de vista corporativo para replantear los servicios o productos que utiliza el usuario y necesitan ser corregidos de acuerdo su verdadera necesidad.
- f2: Pivotaje desde el punto de vista de apoyo de cualquiera de los componentes externos o internos a la organización sean de infraestructura administrativa o tecnológica en beneficio de cumplir los objetivos organizacionales.

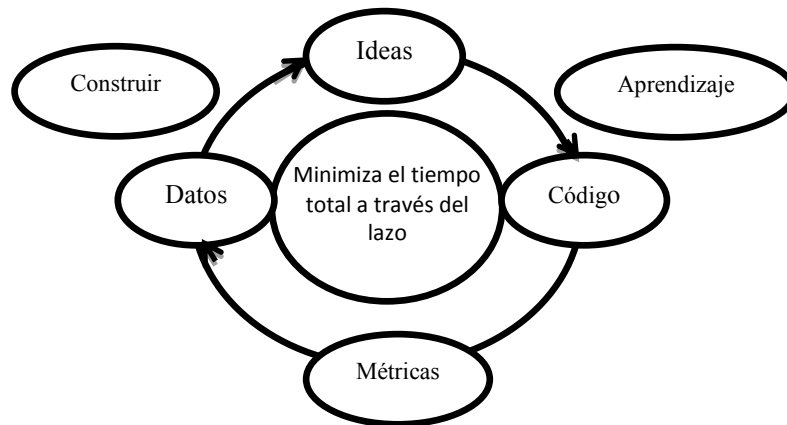


Figura 2.5 Diagrama de Principios de la Metodología Lean Startup

El primer paso es determinar el problema que necesita ser resuelto y luego desarrollar un Producto Mínimo Viable (PMV) para comenzar el proceso de aprender lo más rápido posible. Una vez establecido el PMV, una empresa puede trabajar en afinar el motor. Esto implicará la medición y el aprendizaje y debe incluir métricas accionables que puedan demostrar la causa y el efecto de que se trate.

La propuesta del PMV permite plantear productos mínimos viables desde dos ópticas:

- Productos mínimos viables desde el punto de vista de servicios y ofertas que ofrece la organización que permita un direccionamiento directo con las áreas de apoyo para convertirlas en prestaciones potenciales a los usuarios.
- Productos mínimos viables desde el punto de vista de servicios y ofertas de las áreas de apoyo con relaciones diagonales entre las distintas infraestructuras administrativas y ti para lograr alineamientos directos con los servicios corporativos.

La propuesta de principios de la MLS expuesto en la Figura 2.5, plantea un reto directo entre el nivel ejecutivo y gestión, convirtiendo la estructura organizacional que resultaba complejo en un engranaje simple en base a las ideas e iniciativas de todos los integrantes de la organización, con propuestas de innovación en base a la experiencia administrativa y operativa como aporte al cambio de la alta gerencia.

Para llegar al planteamiento de adaptación de los modelos es necesario partir de los conceptos del Modelo SAM.

2.4. Buenas Prácticas con Servicios ITIL

ITIL es un marco de referencia que especifica un conjunto de recomendaciones para la aplicación de buenas prácticas en la gestión de los servicios de TI (ITIL, 2015).

Se toma ITIL como referencia en vista que se encuentra como estándar de herramienta que aplica las mejores prácticas de TI (Medina & Rico, 2009)(Medina & Rico, 2009).

ITIL tiene el soporte de los siguientes elementos:

- Soporte del Servicio

- Entrega del Servicio
- Administración de Infraestructura
- Administración de Seguridad

ITIL basa su fundamento en la definición del ciclo de vida de servicios enmarcados en estos cuatro elementos.

Provee un enfoque de servicios orientado a la gestión de procesos de TI, diseñados para ayudar a las organizaciones a sostener un uso adecuado de los componentes de TI.

Una de las metas de ITIL es la racionalización de recursos de software y hardware.

ITIL maneja la siguiente estructura de gestión de servicios. (Ver Tabla 2-2) (ITIL, 2015)(Hrabinski, 2009).

Tabla 2-2 Ciclo de vida de los Servicios de ITIL

Soporte del Servicio SS	Función Service Desk	FSD
	Administración de Incidentes	AIN
	Administración de Problemas	APR
	Administración de Cambios	ACA
	Administración de Configuración	ACO
	Administración de Entregas, Versiones y Despliegues	AEV
Entrega del Servicio ES	Niveles de Servicio	ANI
	Capacidad	ACP
	Disponibilidad	ADI
	Continuidad del Servicio	ACN
	Costos de TI	ACS
Administración de Infraestructura AI	Gestión de Operaciones, Instalación y Aceptación	GOI
	Monitoreo	MON
	Automatización	AUT
Administración de Seguridad AS	Operativo	OPE
	Táctico	TAC
	Estratégico	EST

Promueve la mejora continua de los servicios. Esta premisa se encuentra en sintonía con los conceptos que llevan adelante los principios de DevOps de integración, entrega y despliegue.

La Gestión de los procesos de ITIL, se la realiza mediante la definición de servicios por capas. Permite el control de la implantación de las mejores prácticas en las organizaciones.

Cada capa aporta en la gestión de los procesos del desarrollo y despliegue de aplicaciones.

A fin de establecer un manejo adecuado de procesos que intervienen en el desarrollo y despliegue de aplicaciones, se ha diseñado un mapeo con varios procesos orientados a obtener una capa que forme parte de ITIL, que efectivice el uso de DevOps:

- Disponibilidad
- Monitoreo

- Versionamiento de aplicaciones

Como parte de esta capa se incluye al principio de DevOps

- Integración Continua

El mapeo mostrado en la Tabla 2-3, está diseñado para lograr una gestión adecuada en con el uso de los recursos que se presentan en el desarrollo y despliegue de aplicaciones.

Tabla 2-3 Capa de procesos ITIL para efectivizar la gestión del desarrollo y despliegue con DevOps

Disponibilidad
Monitoreo
Versionamiento de aplicaciones
Integración Continua

3. ALINEAMIENTO ESTRATÉGICO ENTRE GOBIERNO CORPORATIVO Y TI, MEDIANTE INCORPORACIÓN DE REQUERIMIENTOS Y SERVICIOS AL MODELO SAM

La evolución constante de las TICs genera en las organizaciones una dinámica difícil de alinear con sus objetivos estratégicos. Esto origina una grieta debido a la falta de alineamiento entre estrategia y objetivos corporativos y la administración de los servicios de TI.

El uso de la tecnología para automatizar los procesos sin esquemas de alineamiento directo con los objetivos empresariales, hace que se desperdicie el gran potencial de TI y se incurra en gastos innecesarios. El área de TI debe entregar prestaciones de valor estratégico e innovador que generen valor agregado a los requerimientos corporativos.

La propuesta de un modelo que permita una relación directa entre el Gobierno Corporativo con TI, que ayude a alinear los servicios y requerimientos empresariales con la infraestructura administrativa y los servicios de desarrollo y despliegue con DevOps que ofrece TI, es el objetivo que propone este capítulo.

Se lo plantea mediante un marco conceptual desde el punto de vista heurístico concerniente a la alineación de los procesos Corporativos con TI utilizando y modificando el Strategic Alignment Model (SAM) de Henderson y Venkatraman.

Al lograr una cadena de valor implementada mediante requerimientos y servicios, con el objetivo de alinear las estrategias del Gobierno Corporativo, del Gobierno de negocio y el TI, se consigue flujos de relación entre el servicio proporcionado y el requerimiento solicitado.

El alineamiento Corporativo con TI que se enuncia toma como base al modelo SAM y utiliza como soporte a la MLS, en base a las siguientes consideraciones:

- 1) Toma como referencia el modelo de SAM estudiado en (Henderson & Venkatraman, 1990) (Avison et al., 2004) (Ojo et al., 2009).
- 2) Realiza una adaptación de SAM en base a servicios y requerimientos, cuyo resultado es (SAM-RS).
- 3) Se considera como soporte la metodología Lean Startup que está diseñada para crear nuevos productos en base a ideas, innovaciones o servicios bajo condiciones de incertidumbre extrema.
- 4) Se plantean los procesos y perspectivas de acuerdo a los ajustes de alineación entre tres dominios vinculados secuencialmente (Ver Figura 3.1).

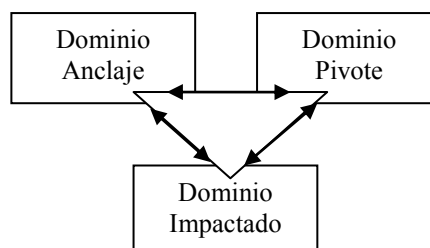


Figura 3.1 Ajustes de alineación entre tres dominios vinculados secuencialmente

3.1. Estructura de Requerimientos y Servicios como apoyo sobre Dominios del SAM

Para realizar el alineamiento estratégico entre los Servicios Corporativos mediante SAM se toma el análisis realizado en (Belalcázar, Díaz, Molinari, 2015), en el cual se aborda la inclusión de servicios y requerimientos en los dominios de SAM.

Es necesario realizar adaptaciones al modelo SAM que permita dinamismo en la identificación de requerimientos y entrega de servicios en la planificación de los dominios encuadrados en sus anclas.

Se parte del concepto de servicio de ITIL que se define como el medio para entregar valor a los clientes. Este valor reside entre otros beneficios, en reducir costos y riesgos como beneficio para el cliente (ITIL Official, 2015).

Habitualmente relacionamos la entidad cliente con el concepto de usuario final. No obstante en una organización, internamente se establecen relaciones que tienen la dinámica requerimiento-ejecución del requerimiento, que son asimilables al concepto de cliente-servidor, manteniendo el concepto que quien es servidor ejecuta un servicio.

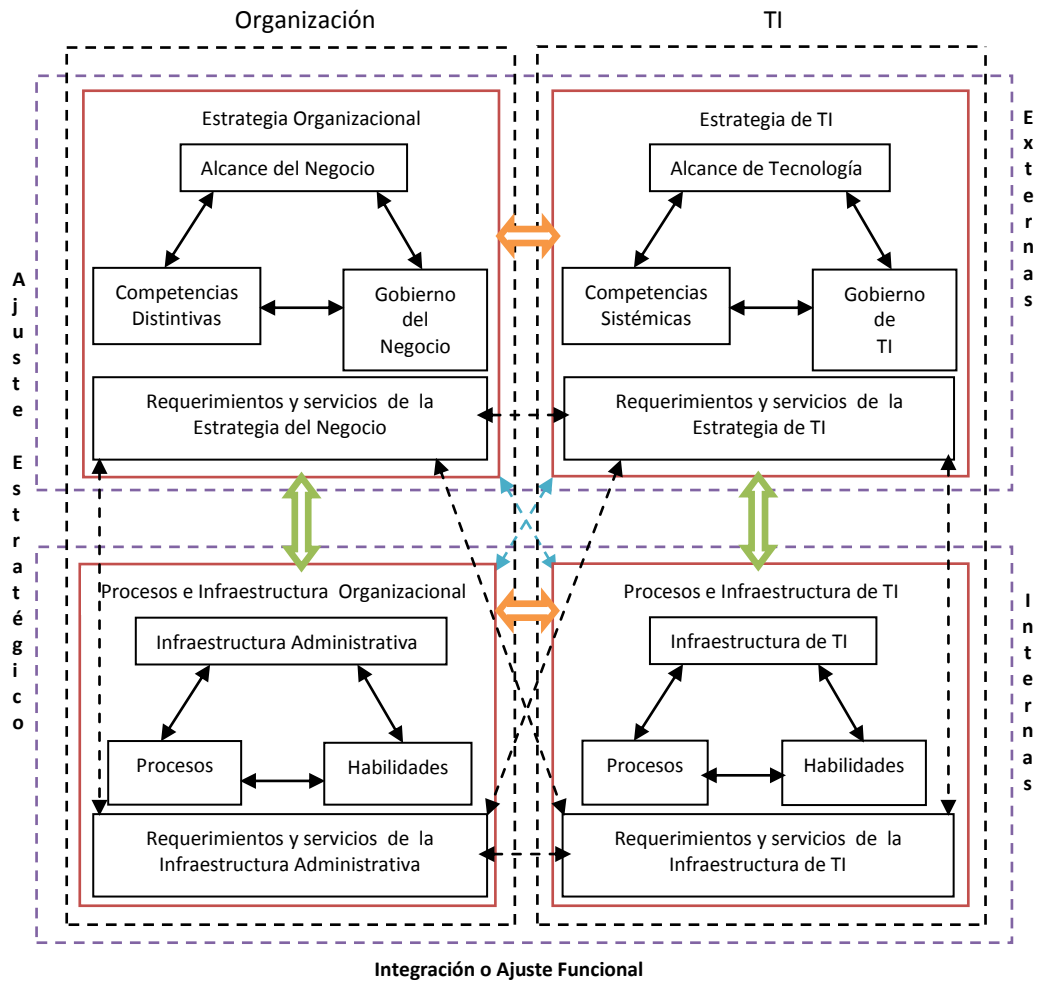
Cuando los accionistas y otros interesados ponen sus expectativas en una organización, estas expectativas pueden ser económicas o con el objetivo de lograr el bien común (en el caso de una organización sin fines de lucro). Cualquiera de estas expectativas, se canalizan en requerimientos que, cumplidos, harán que estas expectativas sean satisfechas.

Esta relación requerimiento-satisfacción no es más que una representación de la relación cliente-servidor. El Gobierno Corporativo se transforma en servidor de accionistas y otros interesados, quien con sus requerimientos se asumen como clientes. El Gobierno Corporativo les ofrece sus servicios corporativos.

A su vez el Gobierno Corporativo realizará sus requerimientos al Gobierno de Negocio en el contexto de su dimensión de conformidad. Estas políticas se canalizan en estrategias. No son requerimientos operativos si no que deben cumplir con las condiciones de conformidad.

El Gobierno de Negocio transformará los requerimientos del Corporativo en requerimientos al Gobierno de TI, en el contexto de competencias distintivas, descritas en párrafos anteriores.

Al insertar requerimientos y servicios en cada Dominio expuesto en la Figura 2.2, no se rompe la relación entre ellos, sino más bien se determina una capa que sirve para reforzar el alineamiento entre dominios con necesidades y prestaciones junto a cada estrategia, como se muestra en la Figura 3.2.



- Clave
- Ajuste o Integración Estratégica
 - Ajuste o Integración Funcional
 - Alineación entre Dominios
 - Relación entre requerimientos y servicios

Figura 3.2 Capa de gestión de dominios en base a requerimientos y servicios

El gráfico anterior mantiene la estructura original de SAM descrita en la sección 2.2.8 y en la Figura 2.2, incluida la estrategia diagonal entre dominios que permite la máxima relación entre ellos ayudando a establecer controles de procesos relacionados (Ojo, Pratap, & Janowski, 2009).

Se adicionan las flechas negras entrecortadas que permiten abstraer la generación de una capa de relación de los diferentes ajustes e influencias ejercidas entre requerimientos y servicios de los cuatro dominios.

Las flechas negras horizontales entrecortadas que unen los requerimientos y servicios de la estrategia del negocio con los requerimientos y servicios de la estrategia de TI, significan que se genera una cadena de requerimientos por parte del negocio (que se constituye en cliente) al Gobierno de TI, y este ha definido su estrategia para dar soporte a las necesidades del negocio.

Las flechas negras horizontales entrecortadas que unen los requerimientos y servicios de la infraestructura administrativa con los requerimientos y servicios de TI, definen a la infraestructura administrativa como cliente que expone sus necesidades de soporte de la infraestructura de TI.

Las flechas negras verticales entrecortadas que unen los requerimientos y servicios de la estrategia del negocio con los requerimientos y servicios de la infraestructura administrativa, definen los requerimientos que solicita la estrategia del negocio a la infraestructura administrativa, esta a su vez provee habilidades del personal y procesos optimizados de acuerdo a la estrategia del negocio.

Las flechas negras verticales entrecortadas que unen los requerimientos y servicios de la estrategia de TI con los requerimientos y servicios de la infraestructura de TI, asumen que el Gobierno de TI se constituye en cliente de sus servicios y de sus procesos e infraestructura. La infraestructura de TI se adapta a las estrategias de TI.

Finalmente las flechas negras diagonales entrecortadas indican que las infraestructuras administrativas y TI proveen servicios a los requerimientos solicitados por las estrategias del negocio y TI, De igual manera la estrategia del negocio y TI proveen servicios a fin de se adecúen las infraestructuras administrativas y TI a los objetivos corporativos.

Los objetivos se canalizan en una cadena de servicios donde los requerimientos que solicita el Gobierno Corporativo tanto al gobierno de negocio como a TI deben ser planificados y justificados en el marco de una estrategia.

El servicio entregado y recibido por cada Dominio conlleva varias aristas de estrategias que se describen a continuación:

- Estrategia del servicio para lograr impacto corporativo.
- Estrategia de optimización de procesos relacionados con el servicio.
- Estrategia para entregar infraestructura de TI adecuada y oportuna a los servicios corporativos y administrativos.

Cada servicio solicitado desde Gobierno Corporativo puede generar varios procesos administrativos y puede involucrar a varios procesos de TI.

El Gobierno Corporativo mantiene su status de control sobre toda la organización en base a los requerimientos que solicita, y que deben ser resueltos por el Gobierno de Negocio y en cadena hacia abajo.

Los requerimientos de los más altos directivos de la corporación hasta los niveles operativos producen procesos de entrada y salida.

Cada gobierno de acuerdo con sus Competencias, establece procesos, actividades y directrices que aporten valor agregado con el fin de conseguir servicios que satisfagan los objetivos empresariales.

3.1.1. Perspectivas o Escenarios de Alineamiento Corporativo y TI con SAM

Cada Dominio tiene su perspectiva para enfrentar el alineamiento Corporativo con TI. Una perspectiva o escenario representa el contexto en que se desarrollará el análisis de alineamiento entre requerimientos y servicios que se provee interna o externamente a un dominio, cuyo objetivo es la satisfacción del usuario.

Las perspectivas establecen los dominios que llevan el control, así como las áreas que apoyan y son modificadas al aplicar los distintos anclajes sobre los dominios involucrados en la propuesta de modernización corporativa.

En base a la Tabla 2-1 se presentan las cuatro perspectivas o escenarios para alinear las estrategias de la Organización y TI (Ojo et al., 2009) :

- (1) Estrategia de Ejecución
- (2) Transformación de Tecnología
- (3) Potencial Competitivo
- (4) Nivel de Servicio

Las cuatro Perspectivas o Escenarios en los Dominios de SAM quedan plasmadas en la Figura 3.3 de la siguiente manera (Ojo et al., 2009):

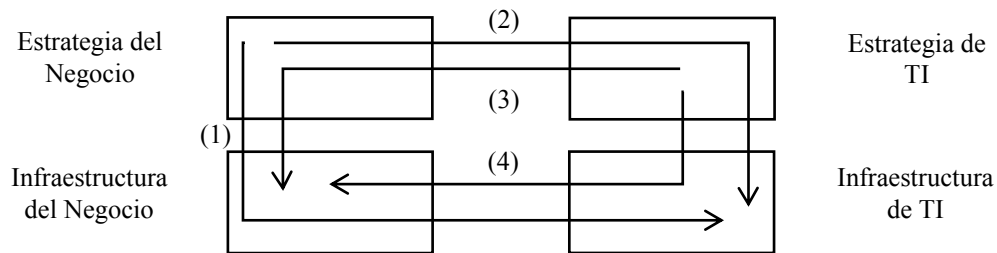


Figura 3.3 Perspectivas de relación de estrategias de los dominios de SAM

En la figura anterior las flechas indican que la estrategia que lleva adelante el control de la modernización de la Organización puede anclarse a cualquiera de los dominios estratégicos del negocio o TI. La visión y decisión de cuál es el dominio que va a enfrentar la transformación o el cambio solicitado por una organización depende de las fortalezas de planificación corporativa y TI, así como del alineamiento de las infraestructuras administrativas y TI.

Para lograr una relación de apoyo ubicua es necesario tomar en cuenta la contribución e innovación de todos los dominios.

Para dar inicio al análisis de las perspectivas de alineamiento corporativo y TI con SAM, se debe partir de las siguientes premisas:

- La estructura de dominios tiene su base en la Planificación Estratégica Corporativa (PE) y en la Planificación Estratégica del Gobierno de TI (GTI).
- Para estructurar los Dominios, tanto PE como GTI deben estar definidas con sus ejes estratégicos, metas y objetivos listos a ser implantados en la organización.
- Si no existe PE o GTI no se puede empezar a determinar los Dominios.

Con el fin de dar una transformación externa e Interna a la organización, se ha establecido el siguiente mapa de dominios en base al flujo de las perspectivas (Ojo et al., 2009):

- 1) Planificación Estratégica Externa del Negocio: PEX
- 2) Planificación Estratégica Interna del Negocio: PIN
- 3) Estrategia Externa del Gobierno de TI: GTIEX
- 4) Estrategia Interna del Gobierno de TI: GTIN

PEX representa la Planificación Estratégica Corporativa y del Negocio que se quiere implantar en la organización (Ver Figura 3.4).

PIN representa la Planificación Estratégica Corporativa Interna preparada por la organización. Brinda apoyo en el establecimiento de modernización de la infraestructura administrativa en la ejecución de la transformación empresarial (Ver Figura 3.4).

GTIEX representa las estrategias externas de TI para dar alineamiento al soporte de la infraestructura de TI con las estrategias y requerimientos planificados por la organización (Ver Figura 3.4).

GTIN representa las estrategias internas de TI para ofertar la infraestructura de TI al servicio de la modernización de las estrategias y procesos corporativos internos acorde a los objetivos empresariales planteados (Ver Figura 3.4).

Las planificaciones de las cuatro estrategias se visualizan en la Figura 3.4.

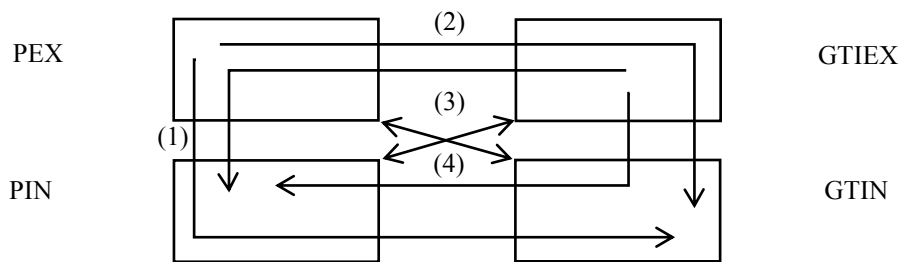


Figura 3.4 Mapa de Dominios con el flujo de las Perspectivas

El apoyo inicial de un dominio a los objetivos empresariales, debe ser con miras a lograr una visión externa e interna de todos los componentes que integran el dominio que está dando apoyo, así como de los dominios relacionados directa y tangencialmente a las estrategias corporativas y de los dominios individuales.

Las relaciones diagonales representan los puntos de interés que tiene cada dominio, sea de apoyo o de estrategia.

Las relaciones diagonales de los dominios son de apoyo entre estrategias de dominios. No intervienen en las decisiones de los dominios que llevan adelante el control de la estrategia de modernización corporativa.

Las relaciones diagonales de los dominios son de apoyo entre dominios operativos. Entregan información de infraestructura administrativa o TI a los dominios que lo solicitan. Apoyan a la estrategia de cada dominio con conocimientos del funcionamiento administrativo y operativo a fin de lograr estrategias con fundamento y soporte hacia objetivos corporativos.

3.1.2. Perspectivas o Escenarios de Alineamiento (1) y (2)

Las perspectivas o escenarios de alineamiento (1) y (2) ayudan a identificar el flujo de control y soporte en la aplicación de estrategias externas e internas de la modernización corporativa (Ver Figura 3.5).

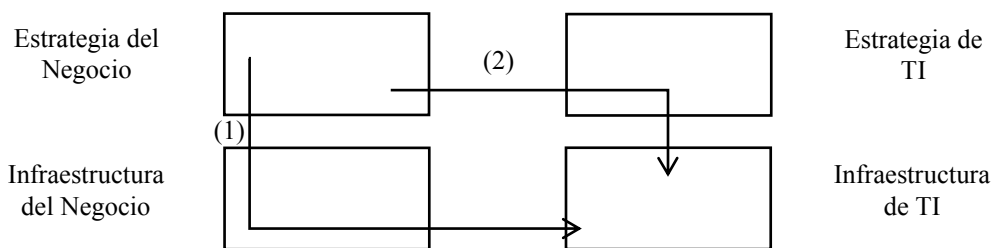


Figura 3.5 Perspectivas de Alineamiento con procesos (1) y (2)

En el escenario (1), la propuesta de modernización surge desde la Estrategia Externa del negocio y el área de cambio es la estrategia interna del negocio, lo que termina afectando a la estrategia interna de TI.

En la perspectiva (1) la organización debe adaptar su planificación interna para dar apoyo con los servicios de infraestructura interna de procesos optimizados, recursos humanos, acorde a los servicios que la organización entrega a sus usuarios. El aporte de servicios que se brinde en esta perspectiva dependerá de las estrategias para relacionarse directamente con la estrategia interna de TI y tangencialmente con la estrategia externa de TI.

En el escenario (2) la Estrategia Interna del Negocio ayuda a transformar la estrategia externa del negocio a través de la perspectiva Nivel de Servicio que entrega la infraestructura administrativa de acuerdo a la estrategia interna y a la estrategia externa de TI, de acuerdo a las necesidades de transformación de servicios que se quiera ofrecer a los usuarios de la organización.

3.1.3. Perspectivas o Escenarios de Alineamiento (3) y (4)

Las perspectivas o escenarios de alineamiento (3) y (4) dan soporte a los cambios en la organización desde la Estrategia de TI cuyo soporte es la estrategia e infraestructura del negocio, y desde la estrategia de TI con el soporte de la infraestructura de TI y de la infraestructura del negocio.

En las perspectivas (3) y (4), la estrategia de TI tiene la oportunidad de lograr cambios y modernizar la organización en base a estrategias de TI y soporte tecnológico alineados a los objetivos empresariales definidos en el plan estratégico corporativo (Ver Figura 3.6).

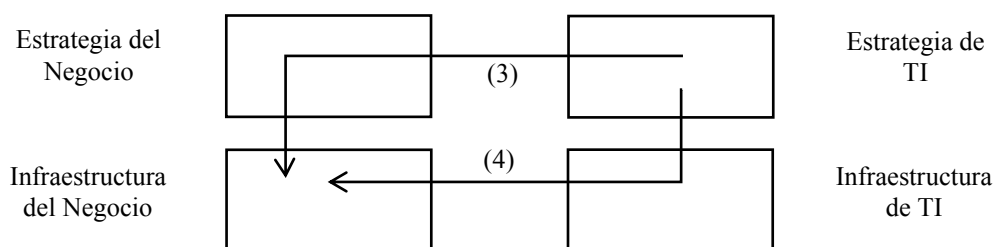


Figura 3.6 Perspectivas de Alineamiento con procesos (3) y (4)

En el escenario (3) la Estrategia de TI da soporte a la transformación interna a través de sus servicios.

En la perspectiva (3) la organización debe adaptar su planificación a los servicios que entrega la infraestructura de TI de acuerdo a la estrategia de TI para dar el soporte adecuado a la perspectiva de potencial competitivo de la organización.

En el escenario (4) la Estrategia de TI, la transformación interna se puede lograr a través de la perspectiva Nivel de Servicio que entrega TI de acuerdo a la estrategia estructurada por TI.

En el escenario (4) la estrategia de TI debe ser la innovación de los servicios de TI adaptados a la infraestructura del negocio que cumpla las expectativas de los objetivos corporativos.

3.1.4. Dominios con Estrategias de SAM de acuerdo a Perspectivas

El modelo SAM de acuerdo a las perspectivas descritas queda planteado en la Figura 3.7 de la siguiente manera:

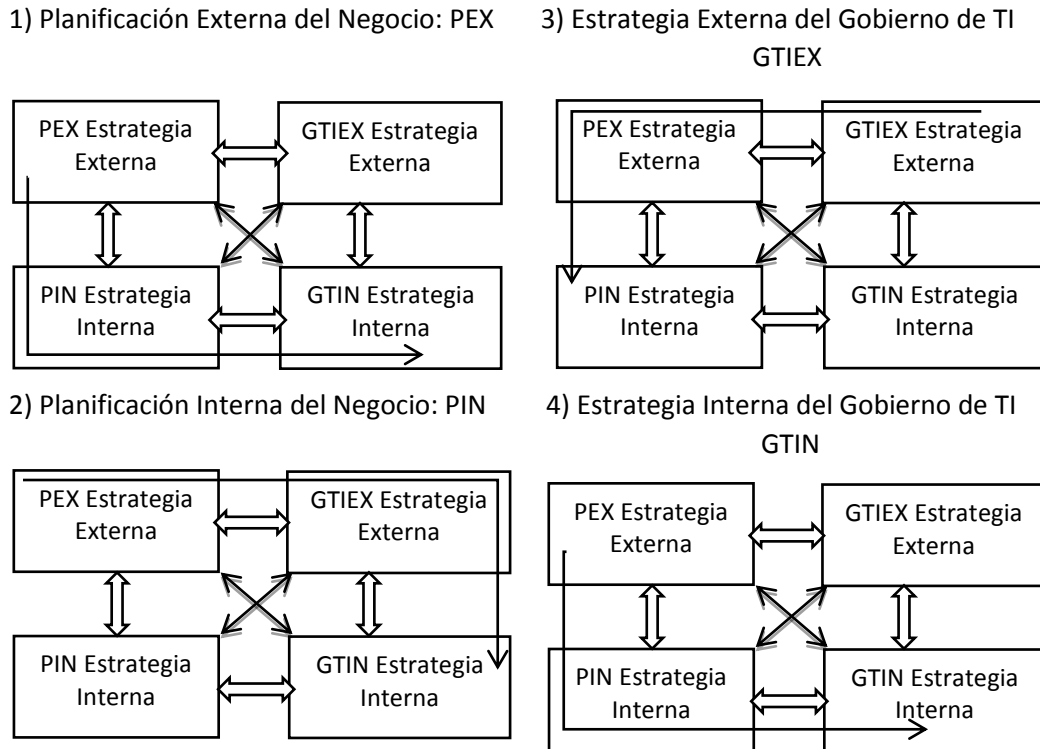


Figura 3.7 Mapa Dominios con Estrategias de SAM de acuerdo a Perspectivas

El mapa de dominios planteado en la figura anterior marca los diferentes escenarios que se pueden presentar en la búsqueda de perspectivas para una modernización adecuada. Las perspectivas estratégicas corporativas dominadas por PEX y GTIEX la responsabilidad de llevar adelante los procesos de control. PIN y GTIN se convierten en procesos de pivote y de apoyo para lograr la mejor estructura de cambio empresarial. Con el fin de dar un mejor ordenamiento y seguimiento a las actividades de cada perspectiva se plantean dos procesos como se muestra en la Tabla 3-1:

Tabla 3-1 Procesos y perspectivas de alineamiento

Proceso	Perspectiva de alineamiento
1	(1) y (2)
2	(3) y (4)

3.1.5. Análisis de procesos con Perspectiva de Alineamiento (1) y (2)

La perspectivas (1) y (2) definen los siguientes dominios (Ver Figura 3.8). El proceso 1 con la perspectiva (1) representada por PEX –PIN - GTIN, y el proceso 1 con la perspectiva (2) representada por PEX – GTIEX – GTIN, servirán para identificar los diferentes procesos, actividades, tareas, requerimientos y servicios para lograr una automatización alineada a los objetivos corporativos.

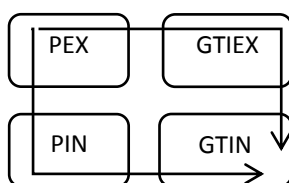


Figura 3.8 Bloque de Dominios y perspectivas (1) y (2)

En la figura anterior se visualiza el alineamiento del Proceso 1 con las perspectivas (1) y (2). Éstas permiten llevar adelante los cambios corporativos mediante el control de La Estrategia de Ejecución del PE y del Gobierno de TI. El apoyo en el Proceso 1 se encuentra fundamentado en la optimización de procesos y el uso adecuado de los recursos de TI.

En la Tabla 3-2 se explica los procesos de entrada y salida que se generan en las perspectivas (1) y (2) de acuerdo a los alcances y competencias de cada dominio, con el fin de lograr el ajuste estratégico, siguiendo un enfoque desde la estrategia hacia los recursos.

Tabla 3-2 Entradas y Salidas de Estrategias de acuerdo a perspectivas (1) y (2)

0	Entrada	Salida	Descripción
Estrategia de Ejecución PE, Proceso de Alineamiento (PEX, PIN, GTIN)			
1	Alineamiento (PEX,PIN)	PIN´	La propuesta de cambio surge desde PEX y PIN adapta su estrategia actual a una nueva y se convierte en PIN´. PIN´ se actualiza en base a los requerimientos del Plan Estratégico Interno con el soporte de la Infraestructura Administrativa para satisfacer lo requerido por PEX.
2	Alineamiento (PIN, GTIN)	GTIN´	La Estrategia Interna GTI se actualiza basándose en la estrategia interna del PE revisada en el paso 1
Estrategia de Transformación de GTI, Procesos de Alineamiento (PEX, GTIEX, GTIN)			
3	Alineamiento (PEX, GTIEX)	GTIEX´	La Estrategia Externa de GTI se actualiza para dar un soporte total a los requerimientos del PE
4	Alineamiento (GTIEX,GTIN)	GTIN´´	La Estrategia Interna de PE se actualiza una segunda vez para apoyar la Estrategia Externa de GTI e indirectamente apoyar al PE.

3.1.6. Análisis de procesos con Perspectiva de Alineamiento (3) y (4)

El bloque de dominios en base a las perspectivas (3) y (4) se plantea en la Figura 3.9. El proceso 2 con la perspectiva (3) representada por GTIEX – PEX- PIN, y el proceso 2 con la perspectiva (4) representada por GTIEX – GTIN – PIN, plantean que GTIEX se encargue de llevar adelante el proceso modernización empresarial. En la explotación de los procesos 1 y 2 se deben establecer los principios de búsqueda de requerimientos y servicios por pares de dominios.

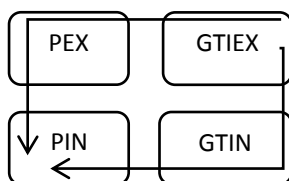


Figura 3.9 Bloque de Dominios y perspectivas (3) y (4)

En la figura anterior se visualiza el alineamiento del Proceso 2 con las perspectivas (3) y (4). Éstas permiten llevar adelante los cambios corporativos mediante el control de la Estrategia de Externa de TI con el soporte de la Planificación interna empresarial. El apoyo en el Proceso 2 se encuentra fundamentado en la búsqueda de información para estructurar esquemas con procesos, requerimientos y servicios alineados a los objetivos empresariales.

Siguiendo la secuencia de perspectivas planteadas en la Figura 3.9 se estructura la Tabla 3-3.

Tabla 3-3 Entradas y Salidas de Estrategias de acuerdo a perspectivas (3) y (4)

0	Entrada	Salida	Descripción
Plan Estratégico PE es manejado por GTI, Procesos de Alineamiento (GTIEX, PEX, PIN)			
1	Alineamiento (GTIEX, PEX)	PEX'	La Estrategia Externa del PE se actualiza para aprovechar las oportunidades basadas en GTI que permita transformación innovadora.
2	Alineamiento (PEX', PIN)	PIN''	La Estrategia Interna del PE se actualiza por segunda vez basándose en las estrategias externas, se aprovechan en forma indirecta las oportunidades que ofrece GTI.
Soporte de GTI, Procesos de Alineamiento (GTIEX', GTIN'', PIN'')			
3	Alineamiento (GTIEX', GTIN'')	GTIN'''	La Estrategia Interna de PE se actualiza para apoyar la estrategia externa del PE. Dado que GTIEX no se ha actualizados desde la última vez que se actualizó por GTIN, el paso es redundante, pero necesario.
4	Alineamiento (GTIN''', PIN'')	PIN''''	La Estrategia Interna del PE se actualiza basada en la Estrategia Interna de GTI.

3.2. Inserción de requerimientos y servicios (SAM-RS) por capas en las Perspectivas de Alineamiento con SAM

La estrategia de alineamiento estratégico en base a inserción de requerimientos y servicios, se lo realiza basado en dos perspectivas:

- Toma como base la estructura y análisis de inclusión de servicios como apoyo sobre dominios de SAM, expuesto en la sección 3.1.
- La inclusión de prestaciones y necesidades por capas en las Perspectivas de Alineamiento con SAM.

Se la realiza basada en la estrategia de identificación de servicios y requerimientos tanto de entrada como de salida de cada uno de los dominios no solo pertenece a los directivos de la Corporación sino a todos los involucrados en los procesos de los otros dominios.

Al insertar requerimientos y servicios a cada una de las estrategias empresariales, se analizan las siguientes pautas:

- La planificación externa del negocio PEX, puede dar paso a una transformación organizacional que defina nuevos requerimientos y conlleven a la estructuración de nuevos servicios, a fin de mejorar sus ventajas competitivas.
- La inserción de requerimientos y servicios están basados en los conceptos de alineación estratégica en base a servicios por dominio, tratados en la Sección 3.1 y expuestos en la Figura 3.2. SAM-RS está definido en la Figura 3.10.

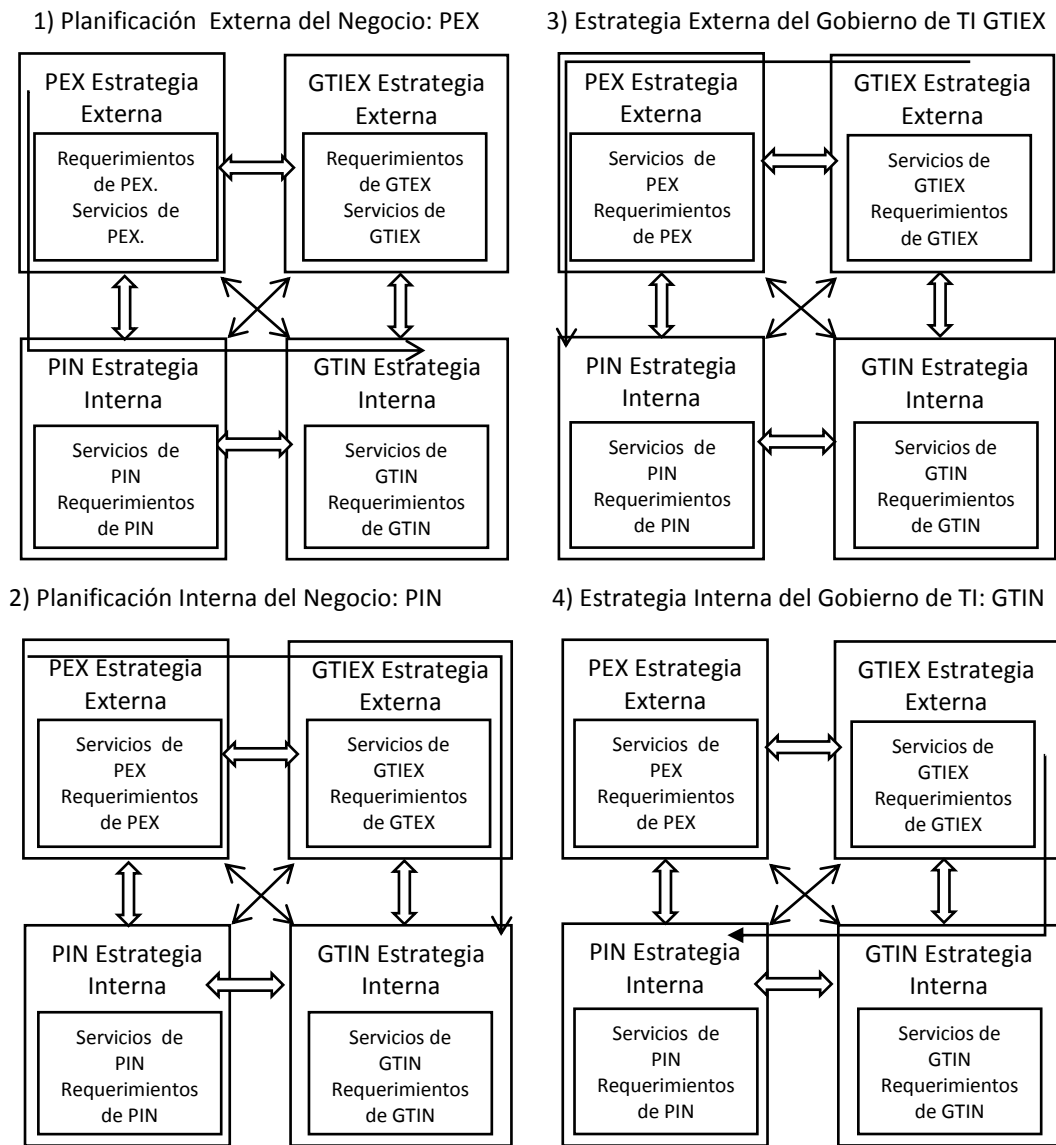


Figura 3.10 Inserción de Servicios y Requerimientos en el Mapa Dominios con Estrategias de SAM (SAM-RS)

Esta nueva capa que se visualiza en la figura anterior llamada SAM-RS, se presenta como resultado de la adaptación que incluye la inserción de requerimientos y servicios realizados a los dominios de SAM.

El modelo SAM-RS abstrae la estrategia externas e internas del negocio y TI como una capa de gestión y operación.

SAM-RS conceptualiza la necesidad de asumir un nuevo modelo y concepción de paradigmas de trabajo.

Ayuda a estructurar relaciones entre el personal corporativo administrativo y tecnológico, cuyas brechas son muy visibles.

En SAM-RS cada Dominio tiene su planificación independiente de acuerdo a los recursos e infraestructura que posee. Cada Dominio al planificar sus estrategias en base a servicios que proporcionan y reciben de los otros dominios, logra visualizar hacia donde se dirigen los requerimientos y a su vez los procesos que intervienen que ya no son exclusivos de cada dominio, ya que interactúan para satisfacer los objetivos internos y externos que les compete.

Las perspectivas de transformación y actualización de planes y proyectos estratégicos de cada dominio, está basado en la identificación de estrategias, requerimientos y servicios que entrega cada dominio como parte de su fortaleza.

Los requerimientos de cada dominio se fundamentan en la necesidad de cubrir cada estrategia de acuerdo a los servicios solicitados por el dominio de entrada.

Los procesos de entrada y salida, así como la infraestructura administrativa y tecnológica requieren estar optimizados y sincronizados, que viabilicen el soporte adecuado y oportuno de acuerdo a los lineamientos del negocio.

La relación transversal de cada dominio depende de la necesidad de recibir o entregar servicios a través de requerimientos que existen o soliciten de acuerdo a la necesidad de cada dominio.

El Pivotaje analizado en la sección 2.3 que es parte de la MLS desde el punto de vista corporativo, replantea el anclaje de los servicios o productos que utiliza el usuario y necesitan ser corregidos de acuerdo su verdadero requerimiento.

Sirve como soporte y complemento conceptual a los procesos y perspectivas de SAM tratados en párrafos anteriores.

Ayuda a definir variantes en la búsqueda de requerimientos y servicios al tratar los procesos relacionados con los objetivos corporativos planteados.

El pivotaje permite abarcar diferentes ópticas de acuerdo al planteamiento de transformación empresarial tenga un dominio.

Como consecuencia de la aplicación del pivotaje se presentan roles y responsabilidades a los cuales no están acostumbrados los actores de cada dominio. Los directivos de la organización enfrentan el reto de entregar un plan estratégico a las áreas y personal de apoyo con el fin de que lleve adelante un proceso de modernización empresarial.

De igual manera las áreas operativas se han acostumbrado a recibir directrices y no tienen la experiencia de llevar adelante un proceso de cambio que tenga como soporte principal las herramientas con las que trabaja en el día a día.

Ayuda a cambiar paradigmas creados en base al status quo, tanto en el ámbito corporativo como operativo.

3.2.1. Alineamiento del Proceso 1 con la perspectiva (1) en base a servicios y requerimientos

En la Figura 3.11 se presenta el alineamiento del proceso 1 con la perspectiva (1). Es la primera representación de alineamiento que debe ser estructurada en dos partes: PEX – PIN y PIN – GTIN.

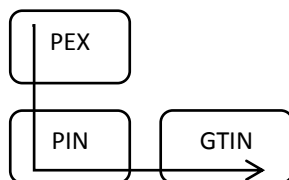


Figura 3.11 Bloque de Dominios y proceso 1 con perspectiva (1) para inserción de servicios y requerimientos

La figura anterior permite abstraer que el Plan Estratégico Corporativo Externo PEX lleva el control de modernización mediante el apoyo del Plan Estratégico Interno y el Gobierno de TI.

La estrategia externa corporativa debe presentar propuestas de modernización de servicios mediante requerimientos hacia las áreas encargadas de la Planificación Interna del Negocio y la planificación estratégica interna del Gobierno de TI.

El soporte hacia la modernización de las prestaciones entregadas al cliente está basado en las estrategias propias del negocio hacia la satisfacción de necesidades del cliente, basados en los recursos que tiene la organización en ese momento.

PIN y GTIN van a variar sus estrategias de acuerdo al grado de cambios que los solicite PEX.

PEX variará de acuerdo a la planificación externa que la organización considere como más apropiada para entregar servicios con valor agregado al cliente.

PEX, PIN, GTIN podrán cambiar sus estrategias de acuerdo como el ciclo de perspectivas lo solicite.

PIN y GTIN son de apoyo directo a PEX.

Las relaciones de estrategias PEX, PIN y GTIN generan entradas y salidas de acuerdo al proceso que se esté estructurando. En la Tabla 3-4 se presenta el proceso 1 con la perspectiva (1).

Tabla 3-4 Entradas y Salidas de Estrategias de proceso 1 y perspectiva (1)

0	Entrada	Salida	Descripción
Estrategia de Ejecución PE, Proceso de Alineamiento (PEX, PIN, GTIN)			
1.1	Alineamiento (PEX,PIN)	PIN'	<p>La PEX del PE toma el control de la modernización.</p> <p>La PIN del PE se actualiza (se genera PIN') en base a los requerimientos del PEX con el soporte de la Infraestructura Administrativa.</p> <p>La Infraestructura Administrativa se alinea en base a los requerimientos del PIN.</p> <p>Se estructuran los servicios que provee la Infraestructura Administrativa como: procesos optimizados, recursos humanos, para definir actividades que ayuden a</p>

			cumplir el soporte a los requerimientos del PIN.
1.2	Alineamiento (PIN, GTIN)	GTIN'	La Estrategia Interna GTI se actualiza basándose en la estrategia interna del PE revisada en el paso 1. La Estrategia Interna GTI define los componentes de Infraestructura que se necesitan para dar apoyo a los requerimientos de PIN.

3.2.2. Alineamiento del Proceso 1 con la perspectiva (2) en base a servicios y requerimientos

En la Figura 3.12 se presenta el alineamiento del proceso 1 con la perspectiva (2). Debe ser estructurada en dos partes: PEX – GTIEX y GTIEX – GTIN'.

PIN' no interviene en este alineamiento directamente pero se podría solicitar su soporte en cualquier momento. En caso de hacerlo se trabaja con la nueva estrategia PIN'.

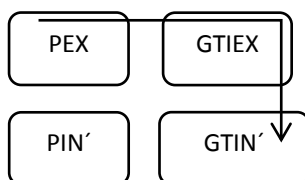


Figura 3.12 Bloque de Dominios y proceso 1 con perspectiva (2) para inserción de servicios y requerimientos

El Plan Estratégico Corporativo Externo PEX lleva el control de modernización mediante el apoyo del Plan Estratégico Externo e Interno del Gobierno de TI.

La estrategia externa corporativa PEX debe presentar propuestas de modernización de servicios mediante el apoyo del Gobierno de TI y este debe proveer apoyo con las estrategias externas e internas de acuerdo a los requerimientos del PEX.

GTIEX y GTIN' requieren apoyo diagonal de PIN', con el fin de analizar las estrategias internas corporativas para encaminar su esfuerzo de acuerdo a los recursos que se vayan a necesitar en la modernización corporativa de acuerdo al PEX.

Las relaciones de estrategias PEX, GTIEX y GTIN' generan entradas y salidas de acuerdo al proceso que se esté estructurando.

En la Tabla 3-5 se presenta el proceso 1 con la perspectiva (2).

Tabla 3-5 Entradas y Salidas de Estrategias de proceso 1 y perspectiva (2)

0	Entrada	Salida	Descripción
Estrategia de Transformación de GTI, Procesos de Alineamiento (PEX, GTIEX, GTIN')			
2.1	Alineamiento (PEX, GTIEX)	GTIEX'	La Estrategia Externa GTIEX de GTI se actualiza para dar un soporte total a los requerimientos del PE. El PE y GTIEX están definidos y conocen los requerimientos y servicios que se necesita proveer para cumplir con los objetivos de satisfacción de servicios al usuario.

2.2	Alineamiento (GTIEX,GTIN')	GTIN''	La Estrategia Interna GTIN' del PE se actualiza una segunda vez para apoyar la Estrategia Externa de GTI e indirectamente apoyar al PE. GTIN' sincroniza actividades de acuerdo a requerimientos internos dados por PIN' y requerimientos externos solicitados por GTIEX.
-----	----------------------------	--------	--

3.2.3. Alineamiento del Proceso 2 con la perspectiva (3) en base a servicios y requerimientos

En la Figura 3.13 se presenta el alineamiento del proceso 2 con la perspectiva (3). Debe ser estructurada en dos instancias: GTIEX – PEX y PEX – PIN', para obtener el máximo beneficio de alineamiento. GTIN'' no interviene en este alineamiento directamente, pero se podría solicitar su soporte en cualquier momento. En caso de hacerlo se trabaja con la nueva estrategia GTIN''.

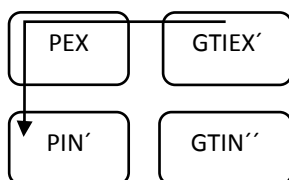


Figura 3.13 Bloque de Dominios y proceso 2 con perspectiva (3) para inserción de servicios y requerimientos

La figura anterior denota que el Plan Estratégico Corporativo es manejado mediante las estrategias externas del Gobierno de TI.

GTIEX lleva el control de modernización mediante el apoyo del Plan Estratégico Externo e Interno Corporativo.

La estrategia externa corporativa PEX y PIN pueden variar y se subordinan a GTIEX.

PEX y PIN se convierten en proveedores de requerimientos hacia GTIEX.

GTIEX plantea las propuestas de modernización de servicios desde su óptica de tecnología.

Una de las ventajas de esta perspectiva es la de verificar las verdaderas necesidades de la organización y los recursos con los que puede aportar para que GTIEX cumpla con su objetivo de modernizar la organización desde su óptica.

De igual manera en esta perspectiva (3) interviene el pivotaje f1, que ayuda a modificar servicios y requerimientos de los dominios, que estaban acostumbrados a mantener su hegemonía como proveedores de servicios externos y solicitantes de servicios óptimos sin participar en las tareas operativas.

PEX debe romper paradigmas para ser parte de los trabajos operativos, con propuestas efectivas de participación económica, administrativa y operativa que contribuya a viabilizar la modernización corporativa independiente del dominio que al frente del proceso de cambio.

Las relaciones de estrategias GTIEX, PEX y PIN' generan entradas y salidas de acuerdo al proceso que se esté estructurando. En la Tabla 3-6 se presenta el proceso 2 con la perspectiva (3).

Tabla 3-6 Entradas y Salidas de Estrategias de proceso 2 y perspectiva (3)

	Entrada	Salida	Descripción
Plan Estratégico PE es manejado por GTI, Procesos de Alineamiento (GTIEX, PEX, PIN)			
3.1	Alineamiento (GTIEX',PEX)	PEX'	La Estrategia Externa GTIEX del PE se actualiza para aprovechar las oportunidades basadas en GTI que permita transformación innovadora. Los servicios que provee GTI a través del GTIEX actualizan el PEX. La visión del nuevo PEX' para satisfacer los requerimientos del usuario puede dar cambios estructurales en la corporación. La adaptación se debe dar debido a que el PE es manejado por GTI.
3.2	Alineamiento (PEX', PIN')	PIN''	La Estrategia Interna del PE se actualiza por segunda vez basándose en las estrategias externas, se aprovechan en forma indirecta las oportunidades que ofrece GTI.

3.2.4. Alineamiento del Proceso 2 con la perspectiva (4) en base a servicios y requerimientos

En la Figura 3.14 se presenta el alineamiento del proceso 2 con la perspectiva (4).

Debe ser estructurada en dos partes: GTIEX' – GTIN'' y GTIN'' - PIN''.

GTIN'' y PIN'' ya fueron actualizados por lo tanto sus estrategias, actividades, procesos están listos para dar soporte a GTIEX'. En caso de existir cambios en sus estrategias se actualizarán de acuerdo a los requerimientos de GTIEX'.

PEX' no interviene en este alineamiento directamente pero se podría solicitar su soporte en cualquier momento. En caso de hacerlo se trabaja con la nueva estrategia PEX'.

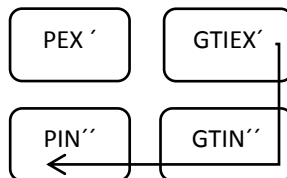


Figura 3.14 Bloque de Dominios y proceso 2 con perspectiva (4) para inserción de servicios y requerimientos

En la figura anterior se abstraer que GTIEX' lleva el control de modernización mediante el apoyo del Plan Estratégico Interno del Gobierno de TI y del Plan Estratégico Interno del Negocio.

GTIN'' y PIN'' se convierten en proveedores de infraestructuras de procesos y recursos computacionales para los requerimientos de GTIEX'.

GTIN'' y PIN'' se subordinan a los cambios que solicite GTIEX' en beneficio de lograr una adecuada modernización corporativa.

GTIEX' plantea las propuestas de modernización de servicios desde su óptica de estrategia externa de tecnología.

GTIN'' debe actualizar su estrategia para que la infraestructura de TI, sea utilizada de acuerdo a los lineamientos que solicita GTIEX'. Los sistemas informáticos, redes y comunicaciones, componentes de Data Center, deben adaptarse a fin de cubrir cada punto del plan de GTIEX'. Es necesario acotar que los cambios en las aplicaciones no son fáciles de resolver, por lo tanto TI debe estructurar estrategias internas para cubrir los requerimientos de acuerdo a su competencia.

PIN'' pone a disposición de GTIEX' y GTIN'' los recursos administrativos que se adecúen a los requerimientos de modernización llevadas por GTIEX'.

Las relaciones de estrategias GTIEX', GTIN'' y PIN'' generan entradas y salidas de acuerdo al proceso que se esté estructurando.

En la Tabla 3-7 se presenta el proceso 2 con la perspectiva (4).

Tabla 3-7 Entradas y Salidas de Estrategias de proceso 2 y perspectiva (4)

0	Entrada	Salida	Descripción
Soporte de GTI, Procesos de Alineamiento (GTIEX', GTIN'', PIN'')			
4.1	Alineamiento (GTIEX', GTIN'')	GTIN'''	La Estrategia Interna GTIN del PE se actualiza una tercera vez para apoyar la Estrategia Externa de GTI e indirectamente apoyar al PE. GTIN sincroniza actividades de acuerdo a requerimientos internos dados por PIN y requerimientos externos solicitados por GTIEX. GTI administra la relación de servicios y requerimientos entre GTIEX y GTIN, en base a los componentes de infraestructura de TI.
4.2	Alineamiento (GTIN''',PIN'')	PIN'''	La Estrategia Interna PIN del PE se actualiza nuevamente basada en la Estrategia Interna GTIN del GTI. El soporte que brinda GTIN con los componentes de GTI, ayuda a buscar innovaciones en el PIN para dar soporte en forma indirecta al PE y provea servicios satisfactorios al usuario.

En la generación de las 4 perspectivas cada una con dos alineamientos, es posible establecer un conjunto de requerimientos y, ligados a estos, están la infraestructura administrativa y TI, con sus procesos y recurso humano, condición suficiente para realizar cambios a las prestaciones que requieren los usuarios.

4. INCORPORACIÓN DE HERRAMIENTAS EN EL MARCO CONCEPTUAL DEL DESARROLLO Y DESPLIEGUE DE APLICACIONES CON DEVOPS

La *automatización de tareas* es una de las metas más deseadas y complejas en esta nueva concepción integradora promovida por las metodologías de desarrollo ágiles y un marco colaborativo entre desarrolladores y operadores.

El grado de congruencia entre las actividades del desarrollo y despliegue de software con los requerimientos del usuario, depende de los mecanismos de alineación de las áreas de desarrollo de software, administración de sistemas y usuarios.

El análisis para identificar los diferentes componentes de los principios de DevOps con su arquitectura basada en integración, entrega y despliegue continuo, así como las fronteras de los roles, logrará regular actividades sobre procesos sensibles, permitiendo agilidad en la entrega de prestaciones al usuario.

En este capítulo, el desarrollo y despliegue de aplicaciones con DevOps se plantea desde varias aristas:

- Se plantean los conceptos de roles y del ciclo de vida en base a metodologías ágiles y en base a DevOps. Se revisa cada fase del ciclo de vida con DevOps basados en sus principios de Integración, entrega y despliegue continuo.
- Se identifican los elementos del despliegue de aplicaciones, incluidas las herramientas que sirven efectivizar la labor colaborativa de DevOps.
- Se incorporan las herramientas a componentes del desarrollo y despliegue de aplicaciones en base a DevOps, tomando en cuenta los criterios de necesidad del negocio de disponibilidad, monitoreo, versionamiento de aplicaciones, e integración continua.
- Se plantea un protocolo para el control de software en producción, a fin de garantizar disponibilidad de servicios a los usuarios, condición indispensable del alineamiento entre la Organización y TI.

4.1. Flujo de tareas o actividades para el caso de nuevas versiones de programas fuentes

Internamente el ciclo desarrollo-testing producción/despliegue puede estar formal o informalmente definido y ser efectivo en cuanto a la determinación de las funciones y responsabilidades involucradas.

Sin embargo, cuando la aplicación se encuentra en producción, el usuario interno o cliente externo puede encontrar fallos o que no cumple con lo esperado por diferentes características que pueden ser la accesibilidad, el tiempo de respuesta, o no tiene opciones que satisfagan la necesidad del usuario. Esos nuevos requerimientos deben ser reportados por los medios que se hayan ofrecido oportunamente desde la organización.

El usuario no es un especialista que puede definir exactamente cuáles son las razones del incidente: si es un problema de software o configuración, si es mala calidad del servicio de Internet en su zona, si lo que se desplegó es una versión obsoleta. Una vez reportado, existen “zonas grises” que dificultan definir quién se hace responsable, si los programadores u operadores. Esta situación de indefinición se manifiesta hasta que la

necesidad de solucionar el incidente conduce, al fin, a la realización de un análisis exhaustivo. Si bien es lo que corresponde, esta iniciativa tardía va a retrasar aún más la respuesta y tendrá impacto en la percepción del usuario acerca de la capacidad de la organización en brindar los servicios.

La detección temprana de los incidentes y una respuesta ágil desde la organización, permite reducir el tiempo para solucionar el incidente y cerrarlo para satisfacción del usuario.

Al no existir una comunicación fluida entre las áreas de desarrollo de aplicaciones y operaciones, se presentan algunos problemas que a continuación se exponen en el siguiente flujo de tareas o actividades para el caso de nuevas versiones de programas fuentes.

- F1: Programador desarrolla la aplicación de acuerdo a requerimientos corporativos.
- F2: Programador realiza pruebas locales y funciona. Se formaliza el ambiente de pruebas de acuerdo a las pautas que se definan entre desarrolladores y dueño del producto.
- F3: Operadores reciben (ya sea en forma manual o automática) el aplicativo desarrollado por el programador y lo publican sin mayor conocimiento de lo que hace el programa.
- F4: Usuario de la aplicación lo utiliza. Encuentra fallos o insatisfacciones.
- F5: Usuario llama a TI con el fin reportar el incidente.

Consecuencias: Considérese que a partir de este momento, la percepción de calidad de respuesta de la organización empieza a ser considerado por el usuario. Está en riesgo el prestigio o imagen de la organización y del área de TI en particular.

- F6: Se dificulta el escalamiento del incidente al no poder determinar responsables entre las distintas áreas intervinientes en el desarrollo y despliegue de aplicaciones.

Consecuencias: Inestabilidad/descontento en el área de TIC, por discusiones, falta de precisión acerca de quienes están directamente involucrados en el incidente. Posibles deserciones de último momento o replanteos de sueldos, etc.

- F7: Solución del incidente por parte de TI: Cuando se considera que el incidente reportado ha sido solucionado, debe informarse al usuario.

Considérese que si la entidad desarrolla una gran cantidad de aplicativos y no existen los recursos adecuados para el pasaje a producción en forma controlada u oportuna (falta de sincronización entre operadores y desarrolladores, diferentes ambientes que configurar) pueden producirse retrasos o incluso la publicación de versiones obsoletas.

La ansiedad por dar soluciones al usuario, hace que se publiquen aplicativos siguiendo una rutina que si bien no es eficaz, es la que se utiliza y recomienda internamente. Se puede visualizar en casos en los que desarrollo necesita despliegue continuo.

Otro punto importante es el control de las versiones de los programas fuentes cuyo repositorio no siempre contiene las últimas versiones de programas.

Para definir una hoja de ruta que permita alinear DevOps-Usuarios, es necesario realizar un análisis conceptual de los componentes que intervienen en el desarrollo y despliegue de software, cada uno en su ámbito con sus características y propiedades.

4.2. Análisis conceptual del Desarrollo de Software

El flujo de trabajo de desarrollo y operaciones en TI expone varios contextos que deben ser contemplados en la búsqueda elementos que los efectivicen.

El desarrollo y despliegue de software requiere considerar la satisfacción de diferentes partes interesadas, entre otros: satisfacer la demanda de los usuarios, requerimientos corporativos, metodología de desarrollo por la que se opte, organización de desarrolladores y operadores en cuanto a roles y actividades (Belalcázar, Díaz, & Molinari, 2016).

La base de la gestión del desarrollo de software es la metodología que se utiliza para desarrollar las aplicaciones.

En esta tesis se opta por metodologías ágiles de desarrollo, debido a la estructura conceptual en la que se fundamentan, basadas en los aspectos colaborativos y en su ciclo de vida con iteraciones cortas aplicadas a las fases de desarrollo y despliegue de software.

Las Metodologías Ágiles basan su fundamento en el manifiesto ágil (Belalcázar, Díaz, & Molinari, 2016).

El manifiesto ágil parte de la necesidad de la entrega temprana y continua de software de trabajo para satisfacer los requerimientos del cliente (Akerle, Ramachandran, & Dixon, 2014).

Considera su principio en base a las siguientes premisas (Ver Figura 4.1). (Hüttermann, 2012)(Cockburn & Highsmith, 2001)(Pardo, Hurtado, & Collazos, 2010) (Ferreira, Noble, & Biddle, 2007).

Las metodologías ágiles introducen un nuevo enfoque de prácticas al conjunto de herramientas de desarrollo (Gimson, 2012).

Estas prácticas ayudan a estructurar con mayor dinamismo las etapas del proyecto mediante los siguientes elementos: lista de requerimientos, programación de a pares, cliente en el lugar, integración continua, refactorización, desarrollo dirigido por pruebas (Gimson, 2012).

A fin de lograr una mejor sintonía con los involucrados en el estudio de metodologías ágiles se realiza una primera revisión de sus componentes:

- Roles
- Ciclo de Vida de metodologías ágiles

Las estrategias de alineamiento entre usuarios, desarrolladores y operadores deben ser coordinadas desde las etapas tempranas de planificación del proyecto.

Para lograr este alineamiento efectivo entre usuarios, desarrolladores y operadores, es preciso granular los componentes del desarrollo y despliegue de Software.

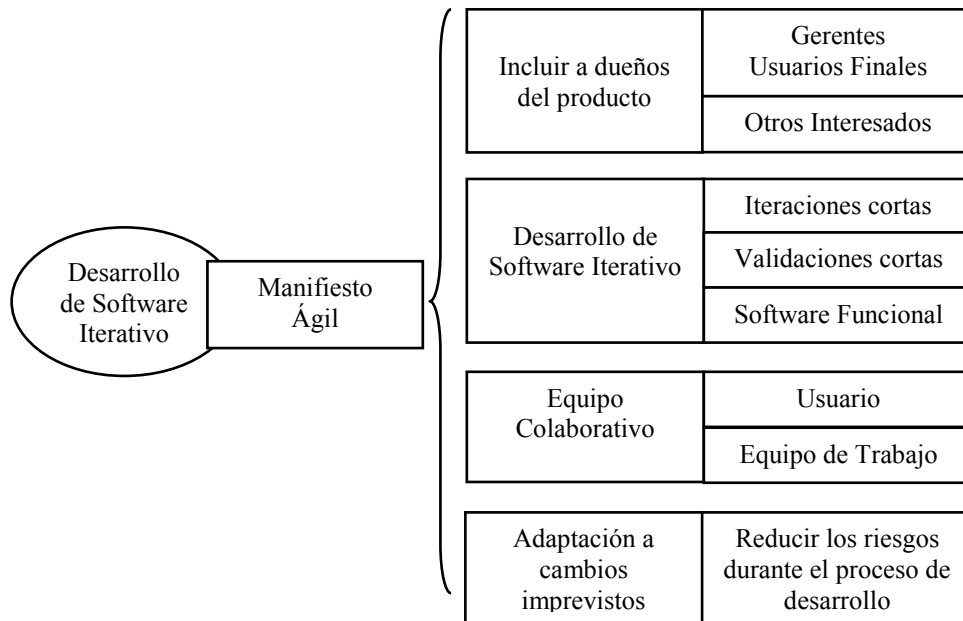


Figura 4.1 Conceptos relacionados de la Metodología de Software Ágil

4.2.1. Análisis de Roles en base a metodologías ágiles

Los Roles que intervienen en las Metodologías Ágiles son los siguientes (Gfader, 2012):

- Equipo de Trabajo
 - Cliente
 - Líder del Proyecto
 - Desarrolladores

El Equipo de trabajo es el conjunto de personas, responsables de llevar adelante el desarrollo de un producto, mediante la compartición de actividades y responsabilidades en cada una de las etapas del desarrollo y despliegue de software.

El Equipo de Trabajo analiza y lleva adelante las siguientes acciones:

- Las tareas a realizar
- Divide el trabajo a realizar en tareas pequeñas funcionales.
- Realiza un análisis para cada área que permita estimar el tiempo que va a tomar el desarrollo de cada una de las tareas.
- Sincroniza tiempos de tareas con equipo de trabajo
- Asigna actividades para desarrolladores.

Las metodologías ágiles impulsan la gestión de desarrollo de software que promueve el trabajo en equipo (Rivadeneira, 2012).

Los equipos ágiles estructuran su propia organización y colaboración, dentro y fuera de los límites de la organización. Los equipos auto organizados no son equipos sin liderazgo; pueden organizarse una y otra vez, en varias configuraciones, para afrontar los retos que se presenten (Cockburn & Highsmith, 2001).

El Cliente aporta con la solicitud de requerimientos y la comunicación de su satisfacción o reclamo.

El Líder del Proyecto es la persona que lleva adelante al equipo de trabajo de acuerdo a una metodología, a reglas y normas que se hayan establecido hacia un objetivo: el desarrollo de software de acuerdo a requerimientos del usuario y del entorno corporativo.

Es el responsable de gestionar los avances y retrasos de las tareas que deben realizar los desarrolladores y ayuda a maximizar los tiempos de respuesta en cada etapa del ciclo de vida del desarrollo de software.

4.2.2. Análisis del Ciclo de vida de las metodologías ágiles

El Ciclo de Vida de las Metodologías Ágiles establece un orden mediante la organización en etapas al trabajo de desarrollo de aplicaciones.

El Ciclo de Vida de las metodologías ágiles se encuentra estructurado como se detalla en la Tabla 4-1.

Tabla 4-1 Ciclo de Vida de las Metodologías Ágiles

Definición de Requerimientos Cliente-DRC
Priorización Tareas-PRT
Planificación Iteración-PLI
Desarrollo-Iteración-DEI
Pruebas Iteración-PUI
Despliegue Iteración-DSI
Control en Producción-COP

4.3. Desarrollo de software en base a DevOps

El desarrollo y despliegue de aplicaciones con DevOps basa su fundamento en las metodologías ágiles que cumplen con satisfacer los servicios de TI que ofrece la organización con un alto grado de automatización. Estos proyectos son aportes de TI a los objetivos corporativos.

DevOps es una mezcla de patrones destinados a mejorar la colaboración entre el desarrollo y operaciones (Hüttermann, 2012)(Humble & Farley, 2011)(Pracht, 2011)(Luthiger, 2015)(Schulze & Adan, 2014).

El objetivo de esta colaboración es mejorar los tiempos de respuesta del ciclo de requerimiento-satisfacción del usuario, mediante metodologías y herramientas que ayuden a alinearlos y cerrar las brechas entre desarrolladores y operadores, aplicando prácticas ágiles en las áreas de desarrollo y operaciones (Rodríguez, Molinari, & Díaz, 2013).

En ese contexto tanto los roles como el ciclo de vida se ven modificados debido a la participación de los operadores como parte del equipo de desarrollo, dando como resultado el equipo de DevOps.

4.3.1. Análisis de Roles en base a DevOps:

Al equipo de trabajo en base a metodologías ágiles definido en la sección 4.2 formado por clientes, líder de equipo, desarrolladores, se adhieren los operadores y forman un equipo colaborativo.

- Equipo de Trabajo en base a DevOps
 - Clientes
 - Líder de Equipo
 - Desarrolladores
 - Operadores

Se incluye en el equipo de trabajo a los operadores ya que DevOps establece la colaboración e integración entre desarrolladores y operadores, constituyéndose de esta manera el equipo DevOps.

4.3.2. Análisis del Ciclo de vida en base a DevOps

DevOps utiliza el ciclo de vida de las metodologías que se encuentra estructurado en la Tabla 4-1 como base para su estrategia colaborativa.

El ciclo de vida de las metodologías ágiles sirve como base para el desarrollo y despliegue en base a DevOps, con una variante en sus 2 primeras etapas.

La responsabilidad de las etapas Definición de Requerimientos del Cliente DRC y Planificación de Tareas PT, le pertenece en su totalidad al equipo desarrollador y no intervienen los operadores, por lo tanto en estas dos fases el equipo DevOps no se encuentra estructurado todavía.

El Ciclo de Vida DevOps inicia en la Planificación de la Iteración PLI y termina en el control de producción COP de las aplicaciones. Sin embargo con el fin de no confundir con el ciclo de vida de las metodologías ágiles se va a tratar las etapas del ciclo de vida ágil como parte del ciclo de vida DevOps y las etapas en las cuales no intervenga el equipo DevOps se dará su responsabilidad a los desarrolladores u operadores con su respectiva aclaración.

4.4. Ciclo de vida en base a DevOps

4.4.1. Definición de Requerimientos Cliente (DRC)

Para lograr el postulado de Satisfacción al Cliente en el cual se basa el Desarrollo de Software Ágil, se parte del análisis de la definición de requerimientos que establecerá las futuras pautas de automatización.

DRC surge de un análisis preliminar entre el usuario y los desarrolladores para definir, entre otros, qué procesos que deben ser automatizados.

En la DRC se logra definir:

- La lista de actividades que luego de ser automatizadas permitan agregar valor a sus labores diarias y a la organización.

- El orden de prioridad de la Lista de Actividades a ser automatizadas
- La agrupación en bloques pequeños de actividades y tareas prioritarias a ser automatizadas
- Las versiones en base a sus requerimientos y prioridades, y en base a la complejidad y velocidad del proyecto.
- Las fechas para cada versión de acuerdo a las prioridades definidas para las actividades a ser automatizadas.

En la etapa inicial de recopilación de requerimientos, la incorporación del usuario al equipo de trabajo de desarrollo de software necesita ser física o presencial.

La participación del usuario y de todos los interesados permitirá que los integrantes del equipo de trabajo logren establecer la demanda real en base a una comunicación fluida.

Las iteraciones cortas permiten evaluar la satisfacción del usuario en forma inmediata.

Las iteraciones cortas permiten abordar soluciones a problemas pequeños que se pueden presentar por falta de una adecuada recopilación de requerimientos.

La relación iteración corta – satisfacción usuario – menor riesgo, se logra a través del aporte del trabajo colaborativo entre el equipo de desarrollo, el usuario y otros interesados.

La presentación de iteraciones cortas permite validar los requerimientos adecuados o no precisos, y permite traer hacia adelante los problemas futuros que se puedan presentar al realizar la entrega completa del proyecto de software.

Al analizar la iteración corta se aborda requerimientos que no están claros para el personal de TIC encargado de la recopilación de necesidades. En este punto este conflicto de requerimientos genera una zona oscura o gris.

La Zona Oscura o Gris es especificada como la etapa en la cual no se tienen claros y definidos los requerimientos del usuario (Hüttermann, 2012).

Un sin número de razones pueden ser descritas entorno a la dificultad que tienen los desarrolladores en recopilar los requerimientos de los usuarios y viceversa, los usuarios en expresar sus necesidades a ser automatizadas.

Diversos factores inciden en un alineamiento preciso entre el equipo de desarrolladores y usuarios:

- Usuario tiene conocimiento del proceso manual pero no tiene facilidad para expresar a los desarrolladores sus necesidades a ser automatizadas, por lo tanto solicita requerimientos erróneos, insuficientes o inconsistentes.
- Desarrollador domina las herramientas de desarrollo pero no tiene habilidades para abstraer los procesos del usuario a ser automatizados
- Desarrollador y usuario no logran establecer una empatía que facilite la recopilación de requerimientos.

La Figura 4.2 ilustra el ambiente de la zona oscura. Mientras no se tengan definidos los requerimientos no se puede dar paso a las siguientes etapas.

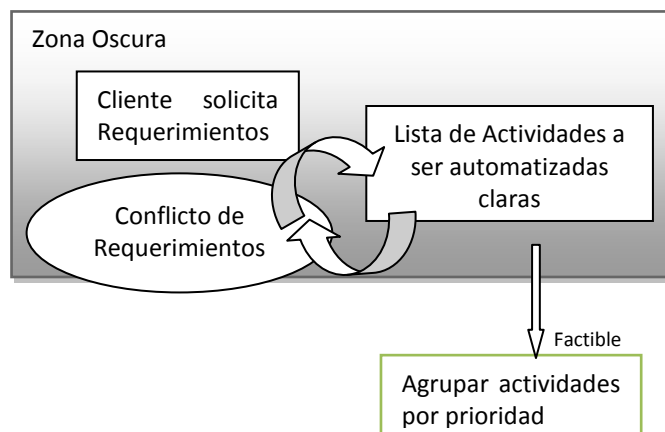


Figura 4.2 Zona Oscura de un Proyecto

En esta tesis se propone un modelo en base a la adaptación de requerimiento y servicios SAM-RS, que permite esclarecer esta etapa.

4.4.2. Priorización Tareas (PRT)

En esta etapa se definen las tareas que son prioritarias para el usuario.

Es en este punto donde se pone en acción uno de los conceptos del manifiesto ágil, que es dar atención a las necesidades reales del usuario.

Las tareas prioritarias son definidas por el usuario.

Las tareas de agrupamiento de requerimientos las realizan los usuarios y el equipo desarrollador.

En la etapa de Priorización de Tareas todavía no interviene el equipo DevOps, ya que no se necesita pre-configuración de componentes del Data Center.

4.4.3. Planificación Iteración (PLI)

El Desarrollo de Software mediante iteraciones cortas ayuda a definir con mayor rapidez la relación aplicativo-requerimiento real.

Una Iteración consiste en un bloque pequeño de requerimientos posibles de ser automatizados en periodos de tiempo de 2 a 4 semanas.

Es necesario establecer una planificación de la iteración que incluya los siguientes aspectos:

- Lista de tareas con su prioridad a ser automatizadas
- Cronograma para cada iteración
- Responsables de cada iteración del equipo de desarrollo
- Responsables de pre-configuración de servicios de Data Center por parte de operadores
- Planificación de herramientas a ser usadas para desarrollo y despliegue de aplicaciones
- Definición de Planificación y responsables de pruebas de las aplicaciones
- Definición de estrategias de despliegue de aplicaciones al usuario

La correcta Planificación de la Iteración da la pauta para un avance adecuado de las actividades que están siendo automatizadas.

DevOps se activa en esta etapa ya que es necesaria la presencia de operadores en el trabajo que llevan los desarrolladores en su colaboración, la es rigurosa en esta etapa.

4.4.4. Desarrollo-Iteración (DEI)

En esta etapa se incluye a los operadores en el equipo desarrollador, por lo tanto se activa el equipo DevOps, cuya labor es el trabajo colaborativo.

Se realizan las siguientes actividades:

- Se planifica la iteración para estimar la complejidad de las tareas a automatizar.
- Se distribuye las actividades de programación entre una o varias personas.
- Se coordina entre desarrolladores y operadores las herramientas de programación, librerías bases de datos, que sirvan de estándar en todo el sistema a desarrollar.
- En cada iteración se realizan varias entregas parciales del producto al usuario. Esto permite que el usuario interactúe con la aplicación así sea en forma de prototipo, lo cual va a ayudar a visualizar las interfaces parciales de sus requerimientos automatizados.

Los operadores coordinan pre-configuración por iteración, esta acción corrige posibles fallos, ya que las configuraciones se van ajustando de acuerdo a la necesidad puntual de la iteración.

En cada iteración que es automatizada por el equipo de desarrollo, se pierde el rol individual de desarrollador y predomina el trabajo en equipo, el cual es el responsable del éxito del proyecto.

En esta etapa se aplica el principio de integración continua. Varios desarrolladores pueden compartir una aplicación, o un proyecto, que puede ser integrado en su totalidad o por partes.

El tamaño del equipo desarrollador puede ser entre 6 y 9 personas. El manejo de menos personal puede comprometer el avance del proyecto debido a los siguientes aspectos:

- Se entrega más trabajo a los otros miembros del equipo.
- Se genera personas indispensables, debido a que no existe personal para reemplazar en caso de ausencia.
- En caso de faltar o renunciar alguno de ellos, se genera inestabilidad en el avance del proyecto.

El éxito de integrar más de 9 personas al proyecto depende de la habilidad del líder del equipo para distribuir las tareas y que estas sean desarrolladas en forma colaborativa.

El equipo DevOps es multidisciplinario, por lo tanto existen aspectos que se deben cumplir para trabajar como tal: cultura, conocimiento, colaboración. Son requisitos indispensables al manejar proyectos de software cuya responsabilidad es de todo el equipo de trabajo.

4.4.5. Pruebas (PUI)

Una prueba es un proceso enfocado a demostrar que una aplicación no tiene errores (Lara, 2013).

La etapa de pruebas ayuda a establecer si los requerimientos automatizados cumplen las expectativas por parte de los usuarios.

Se citan a continuación algunos fundamentos en los que se basan las pruebas (Lara, 2013)(Badgett, Thomas, & Sandler, 2008):

- El desarrollador debe evitar realizar sus propias pruebas
 - En la máquina del desarrollador siempre funciona la aplicación.
Todos los componentes de software como librerías, links están configurados de acuerdo a lo que la aplicación desarrollada requiere.
 - En la máquina del desarrollador todos los componentes de hardware como disco duro, memoria, soportan la aplicación sin ninguna dificultad
 - Parte de las pruebas son los resultados esperados
 - Se realizan pruebas con los datos válidos o normales, es la mitad de las pruebas a realizar
 - Las pruebas deben incluir entradas inválidas e inesperadas
 - Parte de las pruebas son realizadas con datos que no constituyen entradas obvias que ayuden a verificar si funcionan los controles de validación que pueden existir a nivel de campo, de página, de constraint (validación a nivel de base de datos).

Se definen diferentes niveles de pruebas(Lara, 2013)(Badgett et al., 2008)(Rakitin, 2001):

- Pruebas Unitarias
- Pruebas de Integración
- Pruebas de Funcionalidad
- Pruebas del Sistema
- Pruebas de Aceptación
- Pruebas de Instalación

Las pruebas unitarias se focalizan en localizar errores de lógica o de validación en unidades mínimas de algoritmos o códigos sean en funciones, clases, datos.

Las características que dan calidad a las pruebas unitarias son:

- Automatizable
 - Las pruebas de las unidades de código deben ser automatizadas con el fin de que no dependan del desarrollador o usuario
- Reutilizables
 - La creación de pruebas tipo con valor agregado deben ser reutilizadas para diferentes aplicaciones.

- Independientes
 - La ejecución de una prueba en una unidad mínima de algoritmo no debe afectar a las pruebas de otra unidad de código

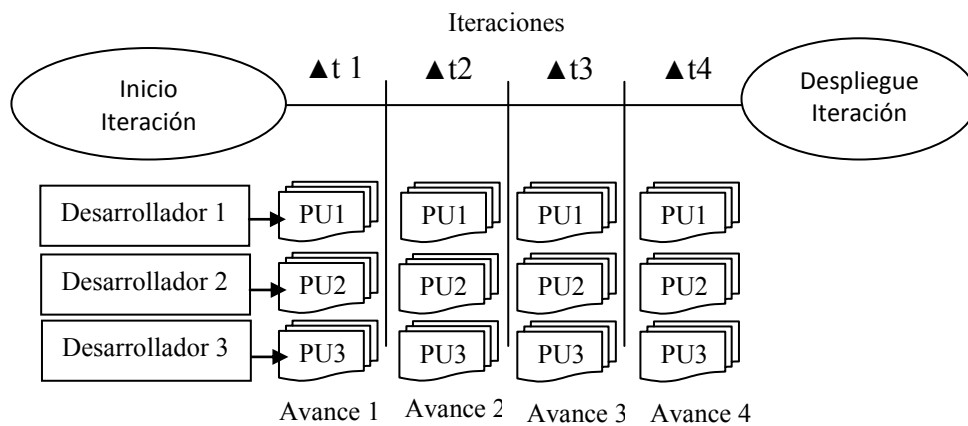


Figura 4.3 Pruebas Unitarias en Desarrollo de Software Ágil

Los $\blacktriangle t$ expuestos en la Figura 4.3 son espacios de tiempo para cada tarea que está dividida la iteración. En cada delta t se realiza en forma simultánea las pruebas unitarias de los desarrolladores.

El desarrollo de software puede ser realizado por varias personas cada una con tareas que luego deben ser unidas mediante interfaces.

Las pruebas de integración se encargan de probar que las interfaces de las distintas aplicaciones que están siendo desarrolladas por separado funcionen en forma coordinada y adecuada.

El éxito de las pruebas de integración se encuentra estrechamente ligado a los estándares de herramientas y procedimientos de salidas y entradas de las aplicaciones a ser usadas por cada uno de los desarrolladores.

En las pruebas de integración interviene todo el equipo DevOps, desarrolladores operadores, y los usuarios como parte esencial de las pruebas.

Las Pruebas de Integración tienen estrategias para realizar su labor:

- No Incrementales
 - Se empieza realizando pruebas en forma individual y luego se realizan las pruebas con todo el sistema en funcionamiento.
 - La lógica de navegación la llevan los responsables de las pruebas en base a lo planificado como inicio y seguimiento de procesos y de acuerdo a la jerarquía de cada módulo.
 - Al ser tratado como un todo se vuelve compleja la identificación de fallos.
- Incrementales
 - Se realiza las pruebas agregando uno a uno los módulos del sistema

Se reconocen 2 tipos de pruebas de integración incremental:

- Ascendentes
- Descendentes

En las pruebas de integración ascendentes los módulos se relacionan en base al nivel de funcionalidad y control que tenga el sistema.

La relación de módulo se estructura en base al diseño de módulo padre y su ramaje de hijos se coordina de acuerdo a la lógica del sistema.

Las pruebas de integración descendentes se realizan partiendo del módulo principal y se mueven en base a la estrategia de diseño que se ha realizado en función de las necesidades de navegación acorde a los requerimientos del usuario.

En estas pruebas los módulos se van incorporando en profundidad o en anchura, de acuerdo a la jerarquía y lógica (Tovar, 2013).

En las pruebas de integración descendentes en profundidad, la integración se realiza por ramas. Cada rama realiza una función específica.

En la Figura 4.4 se presenta un ejemplo con módulos M de una aplicación.

El Módulo 1 (M1) es el principal. Los módulos M2-M3-M4 se subordinan al M1.

M2 tiene a su vez los módulos dependientes M5-M6. M5 contiene a M8.

M3 tiene el módulo M7.

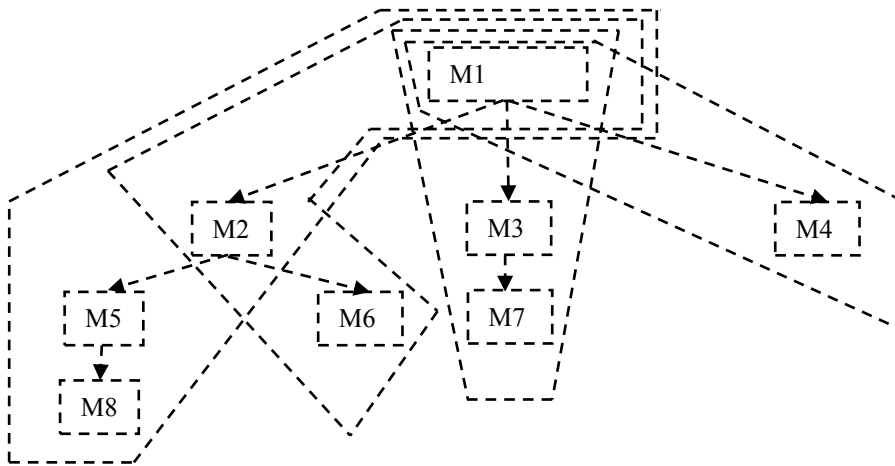


Figura 4.4 Pruebas de Integración Descendentes en Profundidad⁴

Para las secuencias de los módulos M1-M2-M5-M8 y M1-M2-M6 presentados en la Figura 4.4, las estrategias de pruebas son las siguientes:

- El módulo principal M1 tiene parámetros con los cuales se conecta a M2, se realizan las pruebas de funcionamiento e integración con respuestas esperadas para las salidas.
- El módulo 2 invoca al módulo 6 con los respectivos parámetros y se comprueba validez de funcionamiento.
- El módulo principal o el módulo 2 hace una invocación con parámetros al módulo M5, se comprueba resultados en base a requerimientos del usuario y se verifica fallos y aciertos.
- El módulo principal o el módulo 5 llaman al módulo 8 y se ejecuta la funcionalidad de integración

Para la secuencia M1-M3-M7 la estrategia de pruebas es:

⁴ Adaptado de (Tovar, 2013)

- El módulo principal M1 invoca al módulo 3. El módulo 3 se procesa en base a sus entradas propias y a los datos que pasan como parámetros desde el módulo 1 principal.
- Terminadas las pruebas entre el M1 y M3, el módulo 3 invoca al módulo 7 que puede ser llamado también desde el módulo 1.

Para la secuencia M1-M4, la estrategia de pruebas es:

- El módulo principal M1 invoca con parámetros al módulo 4 que trabaja en forma independiente de los otros módulos.
- Se ejecutan las pruebas y se verifican los resultados con los requerimientos del usuario.

La cadena de pruebas que se realizan con estas estrategias, pueden mostrar errores y fallos que pueden ser de diseño o programación de integración, sin embargo todavía no se logra visualizar la relación entre módulos que quedaron huérfanos de pruebas que aparentemente no tienen relación. Para solventar esta relación se tienen las estrategias de pruebas de integración descendentes en anchura.

En las pruebas de integración descendentes en anchura, la integración se realiza por niveles moviéndose en horizontal por la estructura de jerarquía y funcionalidad (Tovar, 2013).

La invocación parte del módulo principal, hacia el módulo que se desea probar. No existe aleatoriedad de llamado ya que la planificación de pruebas permite seguir una secuencia ordenada para controlar la mayor cantidad de posibilidades de ingreso de datos y lograr aciertos o fallos entre la conectividad de los módulos (Ver Figura 4.5).

La aleatoriedad de llamados a los módulos se presenta en la etapa de uso de los módulos en producción por parte del usuario.

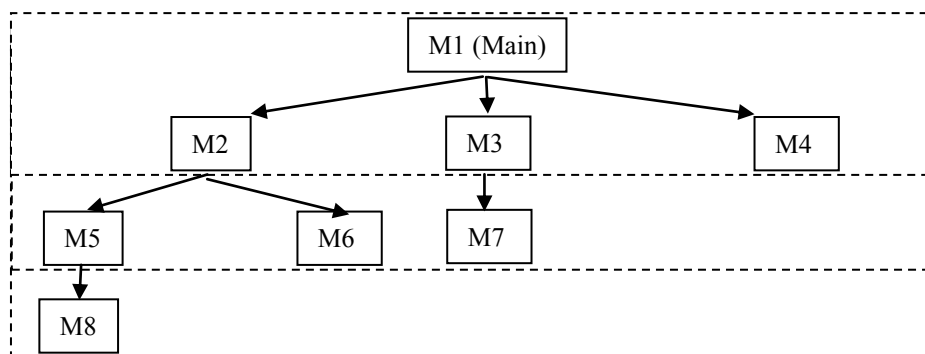


Figura 4.5 Pruebas de Integración Descendentes en Anchura⁵

En las pruebas de integración descendentes en anchura, se realizan las pruebas de los niveles inferiores o subordinados. Se parte de los módulos superiores en base a parámetros y se llega a los niveles inferiores de acuerdo a la invocación que le corresponda. La secuencia de pruebas es como sigue:

⁵ Tomado de Tovar 2013)

Secuencia 1: M1-M2-M3-M4, Secuencia 2: M1- M2-M3-M4-M5-M6-M7, Secuencia 3: M1- M2-M3-M4-M5-M6-M7-M8

Las pruebas de funcionalidad se fundamentan en el análisis, diseño, ejecución, revisión y detección de errores al momento de ser usadas las aplicaciones en su conjunto por el usuario.

El análisis se realiza sobre los datos que se necesitan para realizar las pruebas en base a los requerimientos de los usuarios. Las entradas y procesos que se ejecutan deben ser coherentes con las salidas como procesos, consultas, reportes que den valor agregado a las tareas del usuario.

El diseño se aplica sobre la estructura en la cual están basadas las pruebas. La estructura debe estar diseñada tomando en cuenta la mayor cantidad de posibilidades que tengan los datos a ser probados de acuerdo a cada aplicación, con el fin de lograr la verificación de fallos.

La ejecución sirve para activar los módulos y aplicaciones en base al ingreso de datos en sus respectivos campos.

La revisión se la realiza en base a la navegación y flujo adecuado con el manejo de las pantallas.

La detección de errores se basa en aquéllos que el usuario va detectando en la ejecución de aplicaciones sometidas a la prueba. El análisis de los datos erróneos corresponde al equipo de trabajo con el fin de determinar si es necesario cambios de fondo, o errores de forma que pueden causar molestia al usuario.

Las pruebas del sistema comparan los requerimientos detallados del usuario con los resultados que producen las aplicaciones realizadas por los desarrolladores.

Para realizar las pruebas del sistema, los Requerimientos del Usuario deben estar bien identificados con el fin de hacer un checklist de cada resultado de la prueba con uno o más requerimientos.

Cada iteración de las tareas prioritarias que han sido automatizadas refleja resultados o salidas que deben ser cotejados con las tareas solicitadas por el usuario.

La funcionalidad de todo el sistema se verifica a través de las Pruebas del Sistema en las opciones que satisfagan todos los requerimientos.

Las opciones de aplicaciones que pueden aparecer y que no dan valor agregado a los requerimientos del usuario, son ignoradas para las pruebas del sistema.

Los componentes de las Pruebas del Sistema parten de:

- Planificación de Pruebas a realizar con puntos de control
- Preparación de datos válidos y no válidos

Las pruebas de aceptación son las que se realizan en un ambiente similar al de producción, con el fin de verificar que las aplicaciones desarrolladas cumplan con los requerimientos del usuario.

Los tipos de Pruebas de Aceptación incluyen:

- Pruebas Alfa
- Pruebas Beta

Las Pruebas Alfa son aquellas que son realizadas por el usuario en el sitio en donde se desarrollan las aplicaciones.

El usuario se encuentra junto a los programadores de las aplicaciones en el ambiente de tecnología, donde es capacitado y entrenado para el manejo de las opciones de los sistemas desarrollados.

En este ambiente el usuario prueba y analiza sus requerimientos en base a lo programado. En caso de duda tiene al personal de desarrollo junto a él.

El ambiente de las aplicaciones ya se encuentra disponible en un ambiente de pre-producción, es decir las aplicaciones residen en servidores similares a los de producción.

Las Pruebas Beta son las que se realizan luego de haber concluido las Pruebas Alfa.

Consisten en que el usuario realiza las pruebas del sistema sin apoyo del ningún personal de tecnología.

El ambiente es el lugar de trabajo de los usuarios o clientes y es tecnológicamente compatible con el de producción, tanto en la base de datos, servidores, software de aplicativo.

Las pruebas de ingreso, proceso y salida son ejecutadas de acuerdo a la función que tenga que realizar el usuario, por lo tanto se puede requerir de varios tipos de prueba con varios tipos de personal usuario.

Las Pruebas de Instalación permiten verificar que las aplicaciones se instalan correctamente en las computadoras de los usuarios, tomando en consideración todas las alternativas que se presentan en cada una de las estaciones de trabajo.

Las estaciones de trabajo de los usuarios presentan diversas posibilidades que se controlan a través de las pruebas de instalación:

- Computadores de usuarios con diferente características de hardware: memoria, espacio en disco duro
- Diferentes versiones de sistema operativo y software de escritorio
- Impresoras que requiera distinto tipo de configuración de acuerdo a los controladores
- Procedimientos adecuados con el fin de mantener versiones estables y estandarizadas de las aplicaciones y sus componentes (librerías).

4.4.6. Despliegue de la Iteración-DSI

El Despliegue de la Iteración se basa en la premisa que la Iteración tiene su versión candidata lista para ser usada por el usuario.

En esta etapa se encuentran listos los siguientes componentes:

- Aplicación en base a requerimientos del usuario
- Software y Hardware que requiere la aplicación
- Usuario capacitado en herramientas y aplicaciones a usar
- Redes y Comunicaciones probadas

En la sección 4.6 se profundiza más a detalle los Componentes del Despliegue de Aplicaciones en Data Center.

4.4.7. Control en Producción de la Iteración-COP

Una vez implementada la aplicación, viene la etapa de producción en la que necesario realizar el seguimiento de la aplicación en uso por parte del usuario.

Los cambios en las aplicaciones son tratados en esta etapa como nuevas solicitudes e ingresan desde la etapa de análisis de requerimientos.

El Equipo DevOps continúa dando servicio en esta etapa.

En caso que el cambio sea una solicitud de mantenimiento por no funcionamiento de algún componente del Despliegue en Data Center, el equipo DevOps deberá atender la solicitud en tiempos óptimos.

Si la solicitud de cambio es de lógica de la aplicación, el equipo de desarrollo analiza el pedido y si es viable lo modifica. La aplicación pasa la etapa de pruebas y luego al despliegue.

4.5. Metodología Ágil SCRUM como apoyo a DevOps

El siguiente análisis de la Metodología SCRUM y DevOps se la realiza con el fin de exponer con más claridad y ayuda a entender la relación directa entre Ágiles y DevOps.

Entre las metodologías ágiles ligadas a DevOps están SCRUM y Kamban.

Los roles de los integrantes del equipo ágil en SCRUM están definidos así: Un Dueño de Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master. En Scrum permite a los actores de desarrollo y operaciones a mirar en un solo camino, que es la satisfacción del requerimiento del usuario con entregas funcionales y tecnológicas, listas a ser usadas.

En SCRUM, en la planificación de la lista de requerimientos (Product Backlog), se analiza qué tareas van a ser parte del proyecto. Se realiza junto con el usuario y el equipo de desarrollo (Gfader, 2012). Se identifica la prioridad de cada una de las tareas.

Desde esta fase, el personal de operadores debe formar parte del equipo de Desarrollo. La función del operador es analizar hacia donde se dirige el proyecto y realizar las pre-configuraciones necesarias para cuando termine la Iteración, denominada Sprint en SCRUM.

En la Figura 4.6 se puede apreciar que DevOps empieza desde que los requerimientos representados como ejemplo por R1-R2-R3-R4 ya están definidos para la planificación de cada tarea.

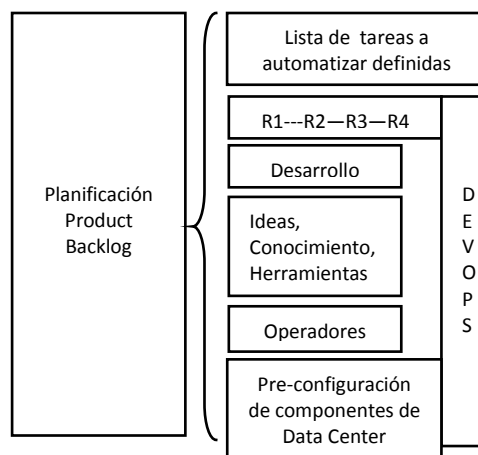


Figura 4.6 Visualización de componentes DevOps en el inicio de desarrollo de software

El Product Backlog lo define el equipo de desarrollo con el usuario. No intervienen los operadores ya que las tareas a automatizar no están definidas.

La definición del Product Backlog no es una tarea sencilla. Dependerá de la habilidad del Scrum Master para lograr un acoplamiento de los requerimientos del usuario con el equipo de recopilación de necesidades a ser automatizadas.

Una vez definido el Product Backlog por parte del Equipo DevOps, se inicia con el desarrollo del Sprint.

La habilidad y experiencia del Product Owner junto con su equipo de trabajo para definir el tamaño del lote o la cantidad de trabajo que se debe realizar por cada sprint es crucial para alcanzar objetivos inmediatos en la entrega continua al usuario.

Una vez terminado el Sprint se somete a test de unidad. En caso de ser válida sin errores, el Product Owner dará el visto bueno para que se lo ponga en producción

Los integrantes o equipo de trabajo de SCRUM tienen los siguientes roles:

Dueño del Producto	Equipo de Desarrollo	Tareas a ser Automatizadas
	Líder Equipo	Scrum Master

Figura 4.7 Roles de los integrantes de una metodología ágil en base a SCRUM

En base a los roles presentados en la Figura 4.7 y utilizando los conceptos DevOps de gestión colaborativa entre desarrolladores y operadores el Ciclo de Desarrollo Ágil de un Proyecto en base a DevOps es el siguiente:

- Equipo de Trabajo
 - Líder de Equipo
 - Usuario
 - Desarrolladores
 - Operadores
 - Tareas a ser automatizadas

El equipo DevOps requiere de una metodología de desarrollo como soporte para lo cual se va a tomar la de SCRUM.

Los componentes de SCRUM son:

- Planificación de Pila del Producto
- Planificación de pila del sprint o tareas prioritarias a ser automatizadas
- Inicio de Iteración
- Desarrollo de Iteración o Sprint
- Obtención de Versión Candidata

El equipo DevOps analiza toda la planificación de pila del sprint de tareas prioritarias a ser automatizadas. En la planificación se asigna tareas a cada integrante del equipo desarrollador y se analiza las herramientas a ser utilizadas. Los operadores que son parte del equipo de trabajo empiezan la pre-configuración de ambientes para alojar las aplicaciones definitivas o de pruebas para el usuario, hasta

Esta versión definitiva que es entregada al usuario también se denomina Versión Candidata.

La Versión Candidata solo pasa a ser tal, cuando es desplegada en un servidor, en un ambiente particular y es utilizada por el usuario sin fallos.

4.6. Componentes del Despliegue de Aplicaciones en componentes del Data Center

Con el fin de establecer el ámbito en el que se desenvuelve el despliegue de aplicaciones, es necesario identificar varios componentes que intervienen en la puesta en producción definitiva de una aplicación:

- Componentes del Data Center
- Granularidad de componentes del Data Center
- Pre-configuración de Componentes de Data Center
- Despliegue de Aplicaciones en el Data Center

4.6.1. Componentes del Data Center

Un Data Center es un espacio utilizado para contener sistemas de cómputo y sus componentes asociados como: servidores, sistemas de almacenamiento, infraestructura de redes y telecomunicaciones (López, 2012).

El Data Center es responsable de los procesos innovadores de las TIC. En este se centralizan las principales tendencias de software y hardware, que tienen impacto en el proceso de desarrollo y despliegue de aplicaciones.

El desarrollo y despliegue de software debe diseñarse en base a los nuevos contextos tecnológicos con infraestructura diseñada para las necesidades corporativas.

El desarrollo y despliegue de software utiliza los recursos del Data Center en forma directa. Para obtener eficiencia y eficacia en la relación de las aplicaciones desarrolladas y desplegadas con los componentes del Data Center es necesario conocer su estructura a fin de precisar en qué tipo de elementos se puede obtener la sincronización más adecuada para ofrecer prestaciones a los usuarios con alta tecnología.

Los beneficios de la relación entre el equipo DevOps con el Data Center son mutuos. En las etapas de desarrollo y despliegue se puede ir configurando aplicaciones con tecnología de punta al cual debe adaptarse el Data Center, y viceversa, servidores con ventajas competitivas obligan a realizar software innovador que utilice estas nuevas tendencias.

Para lograr un alineamiento entre el desarrollo y despliegue de aplicaciones es necesario detallar los componentes del Data Center.

Los componentes de Data Center se pueden clasificar en:

- Componentes Mecánicos y Eléctricos
- Componentes Computacionales
- Componentes de Software

Se identifican componentes que pueden ser configurados de acuerdo a la necesidad de operación, control y monitoreo que tengan los administradores de sistemas.

Cada componente cumple una función específica y en conjunto son el engranaje que sirve para dar servicios finales en las computadoras del usuario.

Los componentes Mecánicos y Eléctricos, son los componentes de hardware que ayudan a mantener el ambiente del Data Center en condiciones propicias para dar continuidad al servicio que se provee. Estos elementos son: UPS, Tableros Eléctricos o PDU, Aire

Acondicionado, Sistema contra Incendios, Seguridades Físicas, Tarjetas de Control, Control de Pasillos Caliente y Frío.

Los Componentes Computacionales, son aquellos que poseen capacidades técnicas y de seguridad que permiten asegurar disponibilidad de operación, entre éstos tenemos: Servidores Tipo Torre, Servidores Tipo Blade, Equipos de Comunicaciones y Firewall, Storage Externo, Firewall físico, Equipos Virtualizados, Servidores Web, Servidores SOA, Servidores de Autenticación, Servidores de Control de Perfiles de Usuario, Dispositivos de Almacenamiento (Sistemas basados en cinta, Unidades de Disco Duro, Unidades de estado sólido, Unidades de Disco Duro Híbrido).

Los Componentes de Software son aquellos programas que sirven para gestionar los recursos del Data Center.

Sirven para controlar la eficiencia y administrar los recursos que se proporcionan al usuario como servicios, mismos que son transparentes ya que solo ven el insumo final que es el sistema informático que utiliza.

En esta categoría se ubica todo tipo de software que va a residir en forma directa en los servidores del Data Center o software que es utilizado en computadores personales que utilicen los servicios de cualquier componente del Data Center, un ejemplo es el software que utilizan los desarrolladores en forma local pero utiliza la base de datos que reside en los servidores del Data Center.

Los servidores del Data Center son computadores que contienen componentes como programas, almacenamiento, memoria, al servicio de otros computadores o personas. En este tipo de componente se almacenan las aplicaciones de los usuarios. Dada la importancia que tienen deben residir en ambientes que ofrezcan seguridad en su alojamiento. Un ambiente seguro en Data Center consta de elementos como UPS, PDU, detectores de incendio, puertas de seguridad. (Ver Figura 4.8).

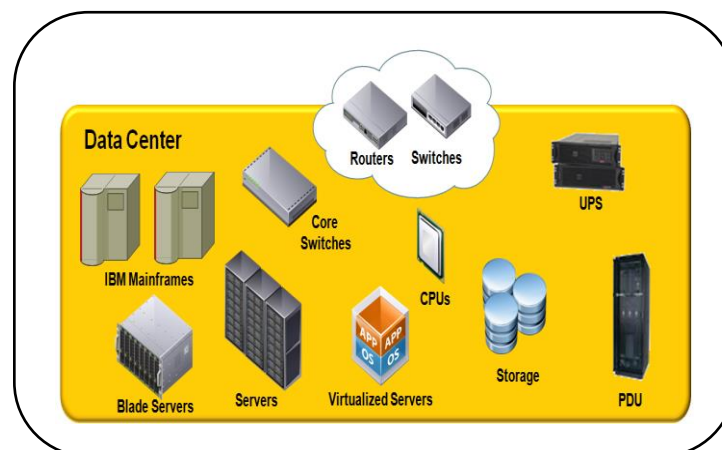


Figura 4.8 Componentes de un Data Center⁶

Los componentes de un Data Center con sus tecnologías, pueden configurar sus sistemas de forma tan detallada que puedan satisfacer las necesidades particulares ambientales y de seguridad con una estrategia de supervisión que puede incluir múltiples puntos de recopilación de datos (Robinson, 2016)(Pracht & Architektur, 2011).

⁶ Figura tomada de Google imágenes

En el Data Center se encuentran los servidores donde están instaladas las aplicaciones y en cadena todos sus servicios relacionados.

No se está considerando Plataforma como Servicio (PaaS), Software como Servicio (SaaS) e Infraestructura como Servicio (IaaS), debido a que se considera que los recursos están instalados y administrados en forma local a la organización.

La garantía de disponibilidad de los componentes del Data Center provee un alineamiento adecuado con los servicios que ofrece la organización a los clientes.

De los componentes en los que se clasifica el Data Center, los de software son los más visibles para reflejar el conjunto de servicios que utiliza el usuario. Por sus características esta tesis aborda este componente en forma detallada, a fin de fundamentar el alineamiento con los objetivos corporativos.

En la Figura 4.9. se presenta un primer flujo básico de los servicios de los componentes que utiliza una aplicación que está siendo usada por el usuario hasta llegar a los servidores del Data Center donde está alojada la aplicación.

El usuario utiliza aplicaciones que han sido desplegadas en los servidores del Data Center. Para llegar a estas instancias la aplicación tuvo que haber sido probada y configurada para establecerse como versión candidata.

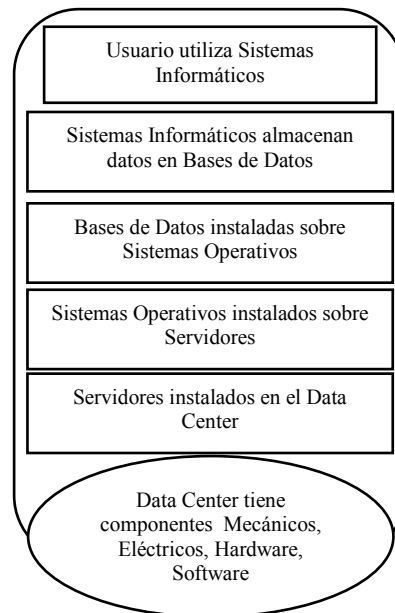


Figura 4.9 Flujo de generación de servicios hasta el usuario

4.6.2. Granularidad de componentes de software

Los componentes de software, son elementos que ofrecen funcionalidades que son diseñadas para ofertar prestaciones que satisfagan las necesidades del usuario.

El código desarrollado se activa mediante varios subcomponentes que puede ser granulados por capas para establecer controles adecuados de funcionamiento (Robinson, 2016) (Pracht & Architektur, 2011).

Para llegar con las aplicaciones hasta el usuario, deben estar activos y disponibles varios servicios que se muestran en la Tabla 4-2.

Tabla 4-2 Servicios activos de Software de Data Center

Servicios activos del sistema operativo
Servicios activos de la base de datos
Servicios del sistema operativo y la base de datos que hacen funcionar a los programas del sistema informático
Servicios de las librerías del sistema operativo para funcionamiento de la base de datos con determinada versión de sistema operativo
Servicios de las librerías de la base de datos para funcionamiento de la base de datos con determinada versión de sistema operativo
Servicios de las librerías del sistema operativo para funcionamiento de los programas del sistema informático
Servicios de las librerías de la base de datos para funcionamiento de los programas del sistema informático
Servicios del sistema operativo para conexión del servidor con el switch core del data center
Servicios de la base de datos para conexión del servidor con el switch core del data center
Servicios del sistema operativo del computador de escritorio del usuario, sobre el cual se instala la aplicación del usuario.
Servicios de las librerías que usa la aplicación usada por el usuario para su funcionamiento en la versión del sistema operativo del computador del usuario.
Servicio de herramientas de ofimática que se relacionan con la aplicación usada por el usuario.

Cada uno de los servicios descritos en la tabla anterior se encuentra activo y distribuido en varias capas que forman las plataformas de software como bases de datos, sistemas operativos.

Es necesario determinar claramente esta distribución de capas para lograr una adecuada relación con las aplicaciones de los usuarios (Ver Figura 4.10). Se analiza al sistema operativo como ejercicio para determinar el camino que sigue una aplicación sobre todos los subcomponentes relacionados.

El sistema operativo se podría entender como la capa intermedia que nos permite interactuar con el ordenador y sobre la que se ejecutan las aplicaciones (G. Fernández, 2007).

Entre las funciones principales del sistema operativo se puede describir en forma general las siguientes: gestión del procesador y la memoria, gestión de los sistemas de almacenamiento, gestión de la entrada y salida, gestión de red, también se puede añadir el sistema operativo provee los servicios a las aplicaciones agilizando su interacción con el hardware. Las aplicaciones a su vez, cuando son instaladas generan sus propios servicios.

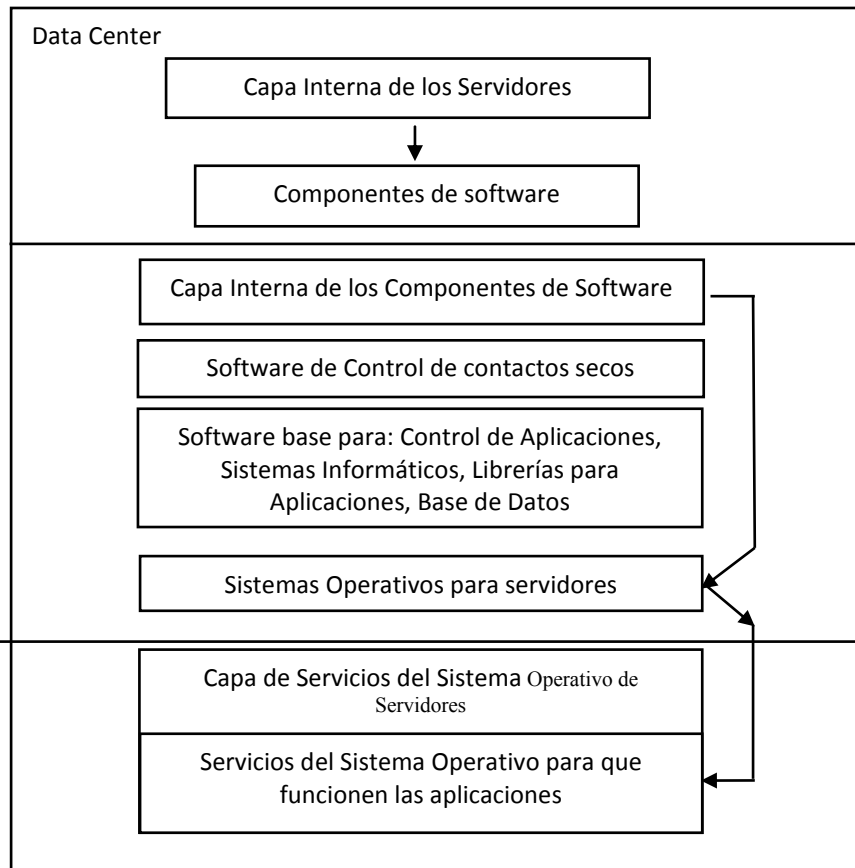


Figura 4.10 Provisión de Servicios del Data Center

Una de las capas internas de los componentes de software es el sistema operativo para servidores y dentro de esta se encuentra la capa de servicios del sistema operativo para que funcionen las aplicaciones.

De acuerdo a la versión del sistema operativo y del software en el que fue desarrollada la aplicación, se debe mantener sincronización de las librerías y componentes relacionados a fin de lograr un adecuado funcionamiento de todos los componentes. Este ramaje debe estar bien definido para lograr la identificación de los subcomponentes de los actores del servicio.

El nivel de detalle de cada capa está dado por la experiencia del personal de TI para obtener el límite de la atomicidad de los elementos configurables en base a la necesidad de convergencia de control y monitoreo que se desea obtener.

Una vez encontrada la relación de cada sistema informático o aplicación con cada uno de los servicios de los componentes del Data Center, se analizan los recursos que son susceptibles de error con el fin de ubicar el monitoreo adecuado y restaurarlo a tiempo antes que se convierta en avería.

En la Figura 4.11 se continúa con el detalle de los componentes del sistema operativo mostrados en la Figura 4.10.

Se presenta el flujo de actores que intervienen al ser usado un aplicativo por parte del usuario. El aplicativo utiliza sistema operativo sobre el cual actúan componentes relacionados con su sistema de archivos y con el aplicativo.

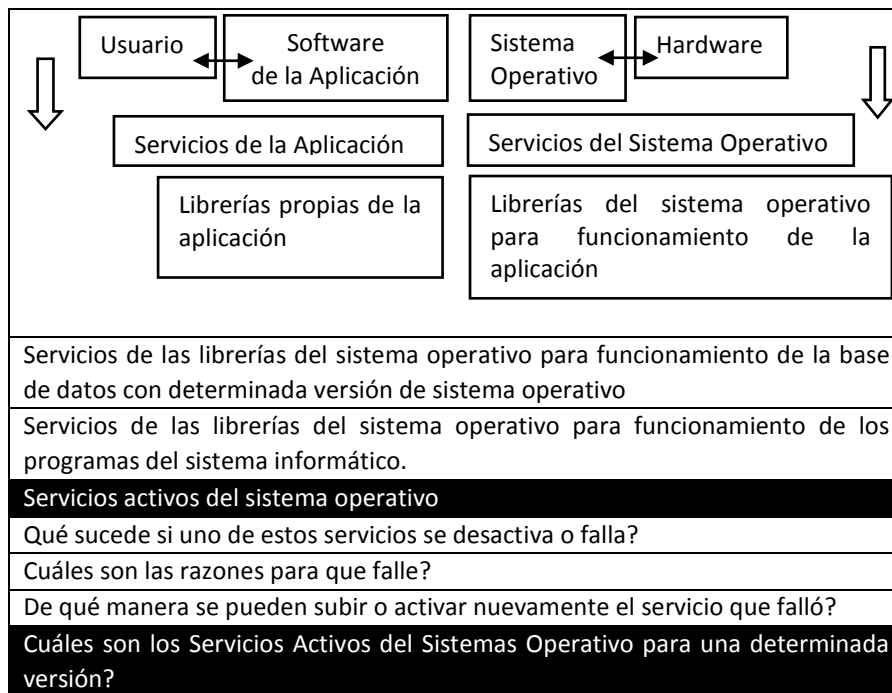


Figura 4.11 Granularidad de Servicio del sistema operativo

En este conjunto de relaciones mostradas en la figura anterior, intervienen librerías propias de la aplicación que aparecen al ser instalada sobre el sistema operativo. De igual manera se activan librerías del sistema operativo para funcionamiento de la aplicación.

Estos componentes tienen la concepción de un servicio, que a su vez tiene relacionados servicios subcomponentes.

Para identificar más en detalle la granularidad de los servicios se toma como referencia al sistema operativo Linux.

Los servicios propios del sistema del sistema operativo deben ser monitoreados ya que si falla alguno de ellos se queda inhabilitado la mayor parte de servicios instalados en el servidor.

En este punto la influencia de MLS mediante su principio de generar PMV es notable ya que las distintas capas que se presentan al granular los componentes del Data Center, ayudan a establecer las mínimas unidades configurables para el control del monitoreo y restauración de servicios.

La relación de los servicios es directa, es decir si falla uno de ellos va a fallar toda la cadena de provisión de servicios.

El ramaje sistema operativo-aplicaciones debe estar bien definido e identificado para lograr la relación de los subcomponentes de los actores del servicio.

El nivel de detalle de cada capa está dado por la experiencia del personal de TI para obtener el límite de atomicidad de los elementos configurables en base a la necesidad de convergencia de control y monitoreo.

La dependencia de las aplicaciones con los servicios de sistema operativo, puede tener varias ramas, por lo tanto se puede volver tediosa la búsqueda de todos sus componentes relacionados. El arduo trabajo que puede representar al inicio mantiene resultados óptimos en la etapa de COP en beneficio del usuario, ya que reduce los errores, y en caso de haberlos logra restablecer los servicios en tiempos cortos.

Como se visualiza en la Figura 4.12 cada aplicativo instalado genera archivos y en algunos casos lo hace como librerías y servicios.

El detalle del sistema operativo se realiza hasta llegar a un máximo probable de control óptimo de sus elementos.

La parte negra sombreada indica la secuencia de actividades que se toman de las Figura 4.10 y Figura 4.11.

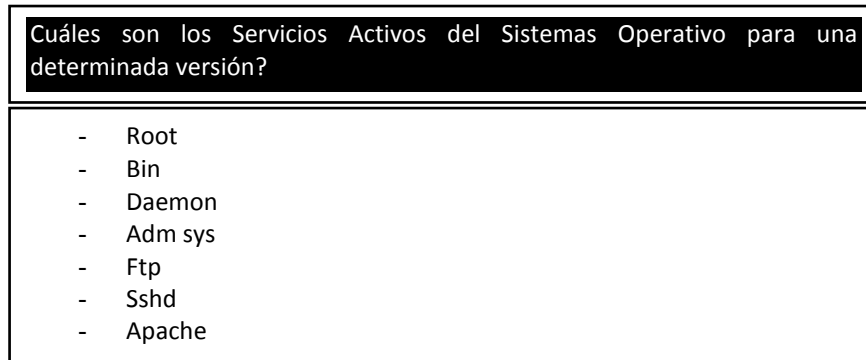


Figura 4.12 Granularidad de Servicio del sistema operativo

Como se visualiza en la figura anterior, el servicio daemon tiene subcomponentes, por lo tanto debemos seguir detallándolos hasta obtener elementos simples controlables.

En el caso del servicio Apache, tiene el servicio httpd, y al ser la mínima unidad configurable. Se configura el control con módulos propios de la herramienta de monitoreo o mediante scripts de sistema operativo.

El tratamiento es similar con los servicios de bases de datos, software de sistemas informáticos, con la consideración que cada uno tiene servicios propios de cada componente y estos a su vez están relacionados con otros servicios como los sistemas operativos.

En el caso de los componentes mecánicos y eléctricos debemos controlarlos con contactos secos y tarjetas de control conectadas a la red de comunicaciones empresarial y si es posible con interfaces de programación que ayuden a enviar mensajes o correos a los operadores de Data Center.

4.6.3. Pre-configuración de Componentes de Data Center

La etapa de Pre-configuración de Componentes de Data Center empieza en la etapa de PLI.

En esta etapa el equipo DevOps coordina las acciones de planificación de las siguientes actividades:

- Elegir herramientas de software a ser usadas en el desarrollo de la Iteración y del Proyecto
- Determinar Frameworks a utilizar en el desarrollo y despliegue de las aplicaciones
- Definir repositorio de Librerías a ser usadas
- Determinar estándar de programación a utilizarse
- Configurar software de escritorio a ser usado por el usuario
- Determinar servidores en los cuales se va a instalar la aplicación

- Definir sistema operativo a ser usado en los servidores de aplicaciones
- Determinar base de datos a ser usada
- Definir tipo de conexión entre aplicativo y base de datos
- Determinar si van a existir aplicaciones para dispositivos móviles
- Definir el tipo de sistema operativo para aplicaciones móviles

En esta etapa el equipo DevOps planifica la estructura de las distintas configuraciones, que van a servir de base para el alojamiento definitivo de las aplicaciones en los servidores.

En razón que la configuración no es definitiva, se la conoce como pre-configuración. Los componentes pueden variar de acuerdo a la necesidad de la programación de la iteración en donde pueden aparecer propuestas de mejora que den valor agregado a los requerimientos del usuario.

En esta etapa se configuran servidores físicos y máquinas virtuales, donde se alojan las aplicaciones.

Las pre-configuraciones se realizan en base al tipo de conexión y aplicación que se realiza: cliente-servidor, ambiente web, bus de servicios, etc.

La etapa de pre-configuración termina el momento en que la aplicación está lista y ya se han definido todos los componentes a ser usados por la aplicación.

En esta etapa, mientras la iteración no se termine, todos los componentes de software y hardware son dinámicos en su configuración.

En la etapa de DEI se determina que, una vez empezado el desarrollo ya no cambian las tareas a ser automatizadas. Sin embargo la especificación de componentes para el despliegue en el Data Center, pueden variar debido a la necesidad de innovación de servicios que se pueda presentar.

4.6.4. Despliegue de aplicaciones en el Data Center

En la etapa de Despliegue de Aplicaciones en el Data Center la iteración o desarrollo de la aplicación se encuentra concluida y se la coloca en los servidores de producción definitivos, para uso del usuario.

Para que el Despliegue de la Iteración tenga éxito se requiere de las siguientes condiciones:

- Que el equipo DevOps cumpla con las etapas de pruebas establecidas con el fin de garantizar que la aplicación está lista para ser usada por el usuario.
- Las herramientas a ser usadas por la aplicación deben estar funcionando acorde a los parámetros establecidos por el equipo de trabajo
- La pre-configuración de la infraestructura debe contar con niveles de soporte que permitan continuidad a las aplicaciones a ser implementadas.

La Versión de la Iteración lista para ser utilizada por el usuario se la denomina Versión Candidata (SCRUM).

El Despliegue de la Iteración se encuentra soportado en el siguiente ciclo de vida (Ver Figura 4.13).

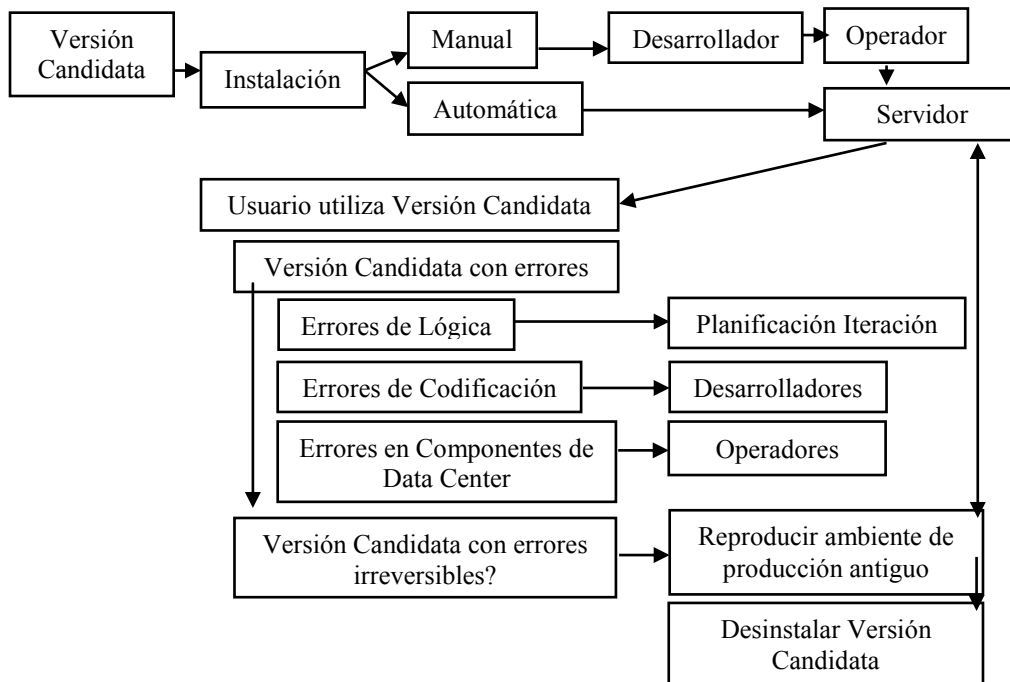


Figura 4.13 Despliegue de Iteración

El flujo de actividades del Despliegue de la Iteración está dada por:

- Versión Candidata
 - o Sin errores de funcionalidad
 - o Pre-Configuración de componentes de despliegue satisfactoria
- Despliegue de Versión Candidata
 - o Despliegue Manual
 - Desarrollador entrega versión candidata a operador para que lo despliegue en el servidor
 - Operador instala en servidor la versión candidata
 - o Despliegue Automático
 - Desarrollador a través de software instala en servidor
 - En esta tarea el operador puede ser comunicado por el desarrollador que ha sido instalada una versión candidata por cualquier medio, chat, correo.
- Usuario ejecuta Versión Candidata
 - o Versión Candidata tiene errores
 - Errores de programación

- Errores de Lógica, es decir la aplicación no cumple con los requerimientos planteados, entonces regresa a la etapa de Planificación de la Iteración.
- Errores de codificación que aparecen al momento de ejecutar la aplicación, entonces regresa al equipo de Desarrolladores
 - Errores de configuración de componentes de despliegue
 - Regresa a revisión por parte de operadores
- Versión Candidata no tiene errores
 - Usuario utiliza la aplicación en forma continua
 - No se reportan novedades

A continuación es necesario identificar los componentes del Data Center donde se despliegan las aplicaciones a ser usadas por el usuario.

4.6.5. Importancia de la identificación de componentes de software del Data Center que utiliza una aplicación

La importancia que tiene identificar los componentes de software del Data Center que están siendo utilizados por una aplicación se lo visualiza mediante el siguiente análisis de un caso ficticio en una situación que se da habitualmente:

- La Organización ofrece servicios al cliente.
- La Organización gasta ingentes recursos en TI para ofrecer servicios informáticos de calidad al usuario.
- TI mediante su equipo DevOps desarrolla y despliega sistemas informáticos estables y probados de acuerdo a los requerimientos del usuario.
- El Usuario o Cliente utiliza los sistemas informáticos sin problemas.
- Los sistemas informáticos son estables durante un tiempo determinado que hace pensar que TI ha realizado un buen trabajo.
- Los sistemas informáticos han sido desarrollados en ambiente web.
- Los sistemas informáticos han sido desarrollados bajo un servidor apache.
- Apache tiene varios servicios que hacen funcionar las aplicaciones del usuario
- Por determinada razón se detiene un servicio de Apache.
- La respuesta de fallo no es atendida en forma oportuna.
- El aparente éxito de TI se vuelve un problema, en vista que usuario necesita el servicio con urgencia.
- Directivos de la organización reciben las quejas de los usuarios al no poder utilizar los servicios en forma continua.
- Directivos insatisfechos con los resultados de TI.
- Al determinar los componentes que son parte del servicio de la aplicación que está siendo usada por el usuario, se puede colocar controles que ayuden a mejorar la disponibilidad del servicio.

Como se puede observar el alineamiento de los componentes de software del Data Center con las aplicaciones que utiliza el usuario, incide directamente en el alineamiento entre TI con la organización, lo que obliga a TI a mantener altos niveles de disponibilidad en sus servicios.

4.7. Incorporación de Herramientas DevOps como apoyo al Desarrollo y Despliegue de Aplicaciones

Esta sección se fundamenta en la revisión conceptual de las herramientas DevOps que sirven de soporte para actualizar y mantener los sistemas de producción estables en todo momento. Estas acciones son uno de los objetivos de la colaboración de DevOps. (Hüttermann, 2012).

El propósito de la incorporación de herramientas es integrar el ciclo de vida DevOps con las herramientas DevOps más adecuadas, en base a criterios de necesidades del negocio: disponibilidad, control de gestión mediante el monitoreo, integración continua y versionamiento de las aplicaciones. Ayudan a establecer fiabilidad en el marco de desarrollo y despliegue de aplicaciones con buenas prácticas.

No es adecuado integrar las herramientas DevOps directamente en cada etapa del ciclo de vida DevOps, es decir definir herramientas para el desarrollo, pruebas o para el despliegue, debido a que se pierde la oportunidad de alinear las herramientas con los objetivos del negocio.

Es habitual que en la infraestructura de TI vayan aumentando la cantidad de servidores a administrar. No obstante, la cantidad de operadores para realizar esta tarea, se mantiene constante. El manejo adecuado de herramientas DevOps ayuda a resolver la administración de sistemas reduciendo los tiempos de desarrollo y despliegue de aplicaciones con el mínimo recurso de operadores (Schäfer, Reichenbach, & Fey, 2013).

Las herramientas a las que se hace referencia permiten ser utilizadas en la etapa de desarrollo en forma exclusiva como Dev (Dev, por Developers, es decir, Desarrolladores), y también en la etapa de despliegue como Ops (Ops, por Operators, es decir, Operadores).

Tanto para Dev como Ops existen herramientas con software, código, scripts o secuencias de comandos que ayudan a controlar de manera eficiente el ambiente de desarrollo y despliegue de aplicaciones.

En esta tesis, a las herramientas tanto para Dev como para Ops se los va a tratar en forma conjunta como DevOps (Relic, 2014).

Las fases de desarrollo y despliegue de aplicaciones requieren de herramientas que permitan a los desarrolladores y operadores realizar sus tareas con efectividad de acuerdo a las responsabilidades y competencias.

Las herramientas DevOps son parte de la estrategia de una buena gestión de administración de recursos. Aportan para cumplir, entre otros, objetivos de automatización de tareas.

En los ambientes de desarrollo de software, estas herramientas soportan integración continua, control de versiones de programas, pruebas automáticas, y despliegue continuo. La gestión de operaciones se implementa mediante herramientas para despliegue automático de aplicaciones, configuración de máquinas virtuales, ejecución automática de scripts.

DRC y PRT son tratados por el equipo de desarrolladores y los dueños del producto. En estas etapas mientras no se logre definir con claridad los requerimientos no se puede avanzar a la etapa de planificación de la iteración (Ver Figura 4.14).

La incorporación de las herramientas DevOps se la realiza desde la PLI y continúa hasta la etapa de COP, en base a las necesidades del negocio (Ver Figura 4.14)..

El Inicio de pre-configuración de componentes de Data Center es desde la etapa de PLI, con el fin de determinar las herramientas que se necesitan para el despliegue de las aplicaciones (Ver Figura 4.14).

El punto permisible para tener completa configuración de componentes de Data Center es la etapa de pruebas. Hasta ese punto se puede variar en la pre-configuración de componentes de Data Center. El momento en que se despliega la aplicación no hay más cambios y la configuración es la definitiva (Ver Figura 4.14).

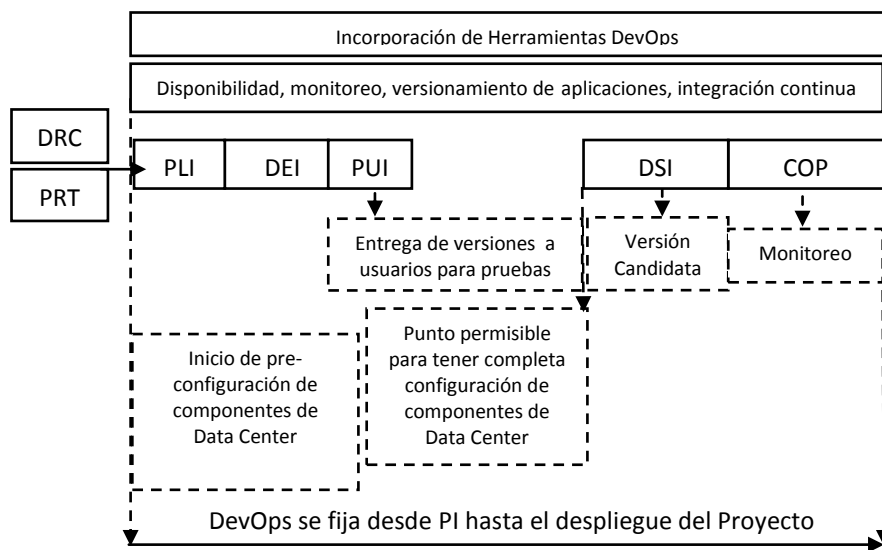


Figura 4.14 Incorporación de herramientas en el ciclo de vida DevOps

DevOps se posiciona desde la PI, como se aprecia en la Figura 4.14.

El equipo DevOps desde PI maneja la integración colaborativa de conocimientos, habilidades de desarrolladores y operadores para alcanzar los objetivos de cada etapa del proyecto.

De esta manera se logra definir anticipadamente los componentes de desarrollo a ser utilizados tales como librerías, características de las máquinas virtuales, pautas de pre-configuración.

Mientras la aplicación desarrollada se encuentre en producción, el equipo DevOps será el responsable de atender los cambios en las versiones de las aplicaciones.

DevOps estructura sus conceptos en base a las tareas y actividades de desarrollo y operadores expuestos en la Tabla 4-3 (Hamaker & Hutton, 2004).

Las responsabilidades y competencias de los desarrolladores y operadores están basadas en los siguientes recursos y servicios que administran desarrolladores y operadores:

Tabla 4-3 Recursos y servicios que administran desarrolladores y operadores

<p>Equipo Desarrolladores:</p> <ul style="list-style-type: none"> - Desarrollo de aplicaciones en base a iteraciones - Integración Continua en desarrollo de aplicaciones - Gestión de distinto tipo de Pruebas de las aplicaciones - Despliegue de Aplicaciones - Gestión de Versionamiento de Aplicaciones <p>Equipo de Operadores</p> <ul style="list-style-type: none"> - Configuración y Despliegue de Aplicaciones. - Configuración y Despliegue de Máquinas Virtuales. - Gestión de Monitoreo de Sistemas Informáticos y Aplicativos - Configuración de Componentes de Software para Sistemas Operativos de Servidor. - Configuración de Software de Bases de Datos Relacionales. - Configuración de Software para desarrollo de Sistemas Computacionales. - Configuración de Software para Servidor de Aplicaciones Java EE de código abierto. - Configuración de Librerías de sistema operativo. - Configuración de Librerías de software de desarrollo - Configuración de Librerías de bases de datos. - Configuración de Servidores - Configuración de Sistemas Operativos de Servidores - Configuración de Respaldos - Gestión de Versionamiento de software de componentes de Data Center - Automatización de tareas de administración de sistemas - Gestión de Monitoreo de Disponibilidad de Componentes de Data Center - Ejecución de procesos batch
--

En la Tabla 2-3 se presenta la capa de procesos ITIL que sirve para efectivizar el ámbito del desarrollo y despliegue de aplicaciones. A continuación se presentan los conceptos que conllevan este criterio de agrupamiento (Ver Tabla 4-4).

Tabla 4-4 Entornos para la correcta administración de desarrollo y despliegue de aplicaciones

Gestión de Disponibilidad	<p>Una Gestión de Disponibilidad adecuada permite cumplir con estos requerimientos (Belalcázar, 2015):</p> <ul style="list-style-type: none"> - Garantizar el nivel de disponibilidad establecido para los servicios de TI - Monitorizar la disponibilidad de los sistemas TI <p>Para garantizar el nivel de disponibilidad de los servicios de TI es necesario tener identificados los componentes a</p>
---------------------------	---

	controlar y aplicar herramientas que ayuden a monitorizar cada componente relacionado con la provisión de servicios de aplicaciones al usuario.
Gestión de Versionamiento	La Gestión de Versionamiento garantiza el control de cambios en el código de las aplicaciones y su software relacionado.
Gestión de Monitoreo	La Gestión de Monitoreo tiene por objetivo controlar la correcta ejecución de los componentes del data center utilizando herramientas que ayuden a detectar a tiempo los fallos que se pueden presentar para ser corregidos y restaurados en forma oportuna.
Gestión de Configuración	La Gestión de Configuración permite gestionar, administrar y controlar los componentes de software tanto en la etapa inicial como en los cambios requeridos.

Esta clasificación sirve de base para clasificar las herramientas DevOps en base a las necesidades del negocio. Se plantea el siguiente esquema:

- I. Herramientas de Desarrollo de aplicaciones con Integración Continua
- II. Herramientas para Gestión de Versionamiento de programas
- III. Herramientas para Configuración de servidores
- IV. Gestión de Virtualización
- V. Herramientas para Gestión de Monitoreo

Se han identificado varias herramientas DevOps que cumplen con esta estructura:

- Para el punto I
 - o Herramientas DevOps de software Jenkins, Perforce
- Para el punto II
 - o Git, GitHub que controlan el versionamiento de los programas y permiten detectar problemas de forma temprana haciendo más fácil el desarrollo de sistemas ya que automatiza procesos repetitivos en un entorno independiente de desarrollo permitiendo ahorro de tiempo.
- Para el punto III
 - o Para la configuración de componentes podemos trabajar con las herramientas OpsCode Chef, Puppet, Cfengine.
- Para el punto IV
 - o Se puede utilizar OpsCode Vagrant, Virtualbox, VMware, Openstack, que ayudan a controlar la virtualización de aplicaciones la cual es estratégica ya que consolida los servidores permitiendo ahorro de energía y más espacios útiles en los equipos.
- Para el punto V

- Para monitoreo podemos trabajar con Nagios, Sensu.

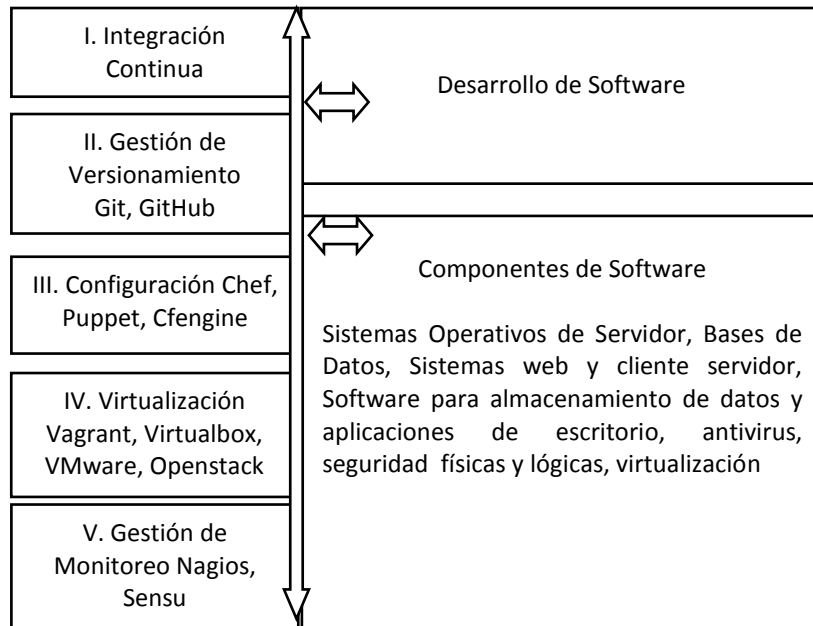


Figura 4.15 Incorporación de herramientas DevOps a componentes del desarrollo y despliegue de aplicaciones con DevOps

El agrupamiento que se visualiza en la Figura 4.15 permite dar una estructura confiable de asignación de recursos a cada componente de Data Center que se refleja en la provisión de servicios que se provee al usuario. (Pracht, 2012).

Se plantea la descripción conceptual de las herramientas relacionadas con el ámbito de DevOps. Cada herramienta maneja su entorno de funcionamiento y puede necesitar de varios conceptos y componentes relacionados, para cumplir con el grupo de herramienta a la cual pertenece.

En ese contexto se presentan las herramientas más representativas, relacionadas en el ámbito de DevOps.

4.7.1. Herramientas de Integración Continua

Las herramientas de Integración Continua permiten integrar varios códigos de aplicaciones realizadas por diferentes desarrolladores en una versión desplegable al usuario. Ayudan a controlar el flujo de trabajo adecuado en el desarrollo colaborativo de las aplicaciones.

Establecen prácticas de desarrollo de software en la que cada miembro del equipo integra su trabajo dando lugar a múltiples integraciones por día.

Cada integración puede ser verificada y probada para detectar errores de integración lo más rápido posible.

Para cumplir con Integración Continua se necesita de los siguientes componentes (González, 2008):

- Sistema de Construcción de Aplicativos
- Servidor de Integración

- Herramienta de Análisis de Código
- Contenedor de Artefactos

Un Sistema de Construcción de Aplicativos es un conjunto de métodos estándar para el desarrollo de software.

Entre los más conocidos sistemas de construcción de aplicativos tenemos a Maven cuyo ambiente de construcción de aplicaciones permite realizar el código fuente sobre Java.

Existen otros sistemas de construcción de aplicativos, como Rake y Ant, cuya característica es similar que Maven ya que permite el desarrollo de software en Java.

En esta sección se la trata debido a su ambiente modular y declarativo, lo cual es estratégico en proyectos complejos.

Maven⁷ es una herramienta para la gestión y construcción de proyectos de software. Ayuda a estructurar, organizar y simplificar los procesos de compilación y generación de ejecutables a partir del código fuente.

Su organización es mediante plugins y librerías.

Las librerías se organizan de acuerdo a la dependencia en cascada que posee, y por lo tanto el tipo de estructura para soportar este tipo de dependencia lo realiza mediante artefactos.

Un artefacto es una estructura que contiene la información necesaria para identificar el grupo, versión, dependencias que tiene la librería padre con otras librerías.

El desarrollador identifica las librerías a ser utilizadas en una aplicación, y Maven las localiza en el repositorio local o remoto al momento de compilar el código fuente.

Es dependiente del ciclo de vida de un proyecto, y no se restringe al trabajo por tareas.

Trabaja bajo el siguiente ciclo de vida con el que se pueda administrar los recursos de código fuente:

- Validación
- Generación de código
- Compilación
- Pruebas
- Empaquetamiento
- Pruebas de integración
- Verificación
- Instalación
- Despliegue de aplicativos

En cada fase del ciclo de vida de Maven, los desarrolladores del proyecto van alimentando la configuración de los componentes de acuerdo como se vaya incrementando la codificación, pruebas, errores y correcciones de los aplicativos del proyecto.

La estandarización de los recursos del proyecto para uso de todos los desarrolladores genera una gran fortaleza ya que disminuye los errores y permite un uso efectivo de componentes necesarios para el éxito del proyecto.

Un Servidor de Integración permite el control y revisión del flujo de tareas desde el inicio del desarrollo de una aplicación por parte de varios desarrolladores hasta el respectivo despliegue. Una de las herramientas que maneja Integración Continua es Jenkins.

⁷ Tomado de <https://www.genbetadev.com/java-j2ee/que-es-maven>

Jenkins⁸ es una herramienta de Integración Continua (CI) que permite:

- Configurar y automatizar tareas periódicas y procesos incrementales.
- Compilar en forma periódica y automática el programa fuente y generar el archivo del proyecto.
- Verificar por espacios de tiempo si se han realizado actualizaciones (Commit) en el repositorio de versiones Git con el fin de compilar el código.
- Coordinación con el software para verificación de errores, los fallos producidos en las compilaciones y su respectivo aviso a los responsables de acuerdo a la comunicación establecida sea por chat, correo.
- Coordinación con software de medición de la calidad de acuerdo a los parámetros establecidos en las variables de configuración de los dos tipos de software.
- Ejecutar los casos de prueba.

La compilación continua de una o varias partes del proyecto conforme se va incrementando el proyecto por parte de los desarrolladores, permite la detección oportuna de fallos, haciendo que sean más fáciles de resolver y automatizar cada proceso individual para luego integrarlos entre todos los programas desarrollados por los programadores.

La calidad del software que se encuentran desarrollando los programadores está ligada a la habilidad de los programadores para trabajar con Jenkins en forma coordinada en el desarrollo de cada aplicativo y luego adicionarlo al proyecto principal.

Las Herramientas de Análisis de Código permiten analizar el código de las aplicaciones desarrolladas en base a convenciones predeterminadas. El software SonarQube permite dar soporte a los desarrolladores en el análisis del código desarrollado.

SonarQube⁹ es un software que permite analizar la calidad del software a través de variables que ayudan a verificar y controlar los siguientes parámetros: errores repetitivos, estándares que se utilizan en el desarrollo.

La parametrización de variables en el software SonarQube dependerá de la necesidad de controlar el tipo de software que se desea desarrollar por parte de los programadores. Se puede obtener demasiada información de análisis del código desarrollado que no es útil, lo cual conlleva a realizar un trabajo infructuoso con métricas que no tienen valor agregado para el análisis del código fuente.

Los parámetros de SonarQube que aportan a los desarrolladores con información para sus análisis son los siguientes:

- Código Duplicado
- Estándares de Codificación
- Pruebas Unitarias
- Complejidad del Código
- Evidencias de Fallos Potenciales
- Nivel de Comentarios
- Valoración sobre Diseño y Arquitectura

⁸Tomado de <https://jenkins.io/>

⁹Tomado de <https://www.infobip.com/es/desarrollador/mejora-de-la-calidad-del-codigo-con-sonarqube>

Al analizar la variable de la herramienta SonarQube que despliega el Código Duplicado se puede verificar las siguientes observaciones:

- Los programadores están trabajando en forma coordinada?
- El código duplicado pertenece a una misma aplicación desarrollada por diferentes programadores?
- El código duplicado pertenece a diferentes aplicaciones?
- Es necesario el código duplicado que se encuentra en diferentes aplicaciones?

El análisis de estas preguntas asegura el trabajo de integración continua.

Los Estándares de Codificación permiten verificar el grado de cumplimiento por parte de los programadores con las reglas establecidas al inicio de la etapa de desarrollo.

La verificación oportuna y en etapas tempranas del cumplimiento de las reglas de programación ayuda a identificar a desarrolladores que pueden poner en riesgo el avance programado definido en la etapa de planificación.

El proceso de pruebas unitarias es clave en el desarrollo de aplicaciones. La Integración Continua depende de la eficiencia con la que se realicen las pruebas unitarias.

SonarQube tiene la posibilidad de parametrizar por desarrollador las pruebas sobre toda la aplicación o parte de ellas.

La cobertura de las pruebas unitarias debe poner énfasis no solo en los procesos complejos, también se lo debe realizar en componentes de programación trivial o sencilla donde pueden existir fallos en código que no se considera esencial.

SonarQube ayuda a encontrar Fallos Potenciales, lo cual evidencia falencias en el tipo de programación de los desarrolladores o en la configuración de variables como librerías o bugs de las herramientas usadas tanto para la programación como la base de datos usada.

La parametrización y estandarización de comentarios ayuda a establecer rutas de soporte adecuado en las diferentes etapas de programación incluidas las de mantenimiento.

La definición para establecer la Valoración sobre Diseño y Arquitectura, ayudan a verificar que la Planificación del Proyecto es la adecuada, más aun en el inicio del proyecto.

La configuración y funcionalidad óptima de cada variable de SonarQube depende de los siguientes recursos disponibles:

- Capacitación de personal
- Recursos Tecnológicos
- Recursos Híbridos

Para realizar el análisis del software que se está desarrollando es indispensable disponer de personal capacitado para revisar cada una de las variables que se han configurado.

Los Recursos Tecnológicos disponibles como almacenamiento, hardware, software, redes, comunicaciones, para albergar toda la información que se va desplegando, entre otras, el log de todas las variables configuradas, deben ser consistentes en su capacidad que permitan disponibilidad de la información almacenada.

Los recursos híbridos son aquellos que tienen que ver con el equilibrio que se establece entre la capacidad del personal para revisar cada una de las variables que se han

configurado con SonarQube y con el apoyo de los recursos tecnológicos que permiten lograr resultados eficaces en el análisis de la información.

Un Contenedor de Artefactos permite el alojamiento de librerías utilizadas en el desarrollo de software. Las librerías usadas en el desarrollo de software se alojan en el repositorio local de Maven, tanto en la etapa de instalación o compilación.

Nexus Oss¹⁰ es un contenedor de artefactos, y es parte del concepto de integración continua. Ayuda a establecer y organizar y compartir grupos de librerías entre todos los involucrados en el desarrollo de las aplicaciones en los entornos de preproducción y producción.

Al tener un contenedor en cada máquina de los desarrolladores, se puede cometer errores de actualización de librerías. Para evitar este tipo de inconsistencias, se centraliza las librerías en contenedores que ayuden a gestionar cada contenedor. De esta manera, si existen actualizaciones, se las realiza una sola vez en el contenedor de Nexus Oss al cual acceden todas las aplicaciones que están siendo desarrolladas por cada programador.

4.7.2. Herramientas de Gestión de Versionamiento

La Gestión de Control de Versiones ayuda a realizar un manejo adecuado de los cambios que se realizan en las aplicaciones.

Git¹¹ es un software que permite el control de cambios que se realiza en el software de las distintas áreas de TIC.

El control automático de versiones que se realiza con Git permite a los desarrolladores un manejo adecuado de aplicaciones en el tiempo.

Git ayuda a organizar las versiones de los programas fuentes y resuelve muchos problemas que los desarrolladores tienen en el manejo de los cambios en las aplicaciones. Entre estos problemas tenemos:

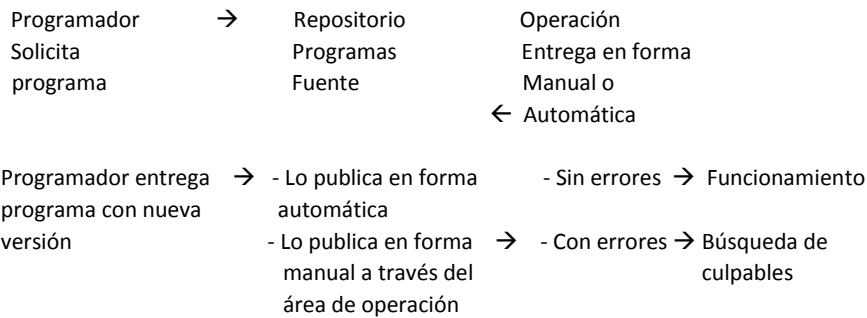
- En ocasiones los desarrolladores mantienen los programas fuentes en sus computadoras personales como repositorio y no en servidores como repositorio central.
- Existe una marcada división entre desarrolladores y operadores al publicar una aplicación en producción. Los desarrolladores publican las aplicaciones sin conocimiento de los operadores. En caso de existir algún error en la aplicación al ser usada por el usuario, el operador devuelve al equipo de desarrollo sin la información de causa raíz que se necesita para reproducir y resolver problemas.
- El personal de desarrollo publica en los servidores de producción las aplicaciones con fallas, sin embargo en sus computadoras locales las pruebas se ejecutan en forma satisfactoria, produciendo descoordinación y desempeño deficiente, lo que afecta a los usuarios.
- Las versiones de software para ejecución de los programas que utiliza el usuario no son las mismas o no son compatibles con las versiones del software instalado

¹⁰ Tomado de <https://www.sonatype.com/download-oss-sonatype>

¹¹ Tomado de <https://git-scm.com/>

en los servidores y en los equipos de escritorio que utiliza el personal de desarrollo para sus aplicaciones.

- El repositorio de programas fuentes se encuentra centralizado en los servidores pero sin control de versiones.
Cada vez que un programador necesita realizar un cambio en los programas, solicita al área de operadores o administrador de Sistemas sea en forma manual o automática el programa a ser modificado. El programa es entregado al desarrollador, el cual lo modifica sin embargo esta nueva versión de programa al ponerla en producción pueden ocasionar errores que en muchos casos no son admitidos por ninguna de las partes, programadores y operadores, el círculo empieza hasta que una de las partes analice a detalle el porqué de los errores y lo transmita al dueño del proceso.
- Al no existir armonía entre las áreas de desarrollo de aplicaciones y operaciones, se presentan algunas posibilidades que a continuación se exponen para el caso de versiones de programas fuentes:



La estructura de Git ayuda a mantener un control adecuado de este tipo de problemas. Cabe anotar que la configuración de Git por parte del personal técnico debe ser acoplada a la planificación de versiones que debe existir en la etapa de Planificación de la Iteración de las Tareas a ser automatizadas. Es decir la planificación del modelo de versiones que se debe controlar, debe estar adaptado a la gran variedad de tipo de aplicaciones que se maneje y al valor agregado que se desea dar al manejo de versiones en corto y largo plazo.

El funcionamiento de Git es a través de imágenes que toma en un momento determinado de los archivos relacionados con un proyecto. En caso de no existir un cambio en los archivos relacionados con el proyecto no saca ninguna imagen y solo almacena el enlace al archivo anterior similar al que ya tiene almacenado.

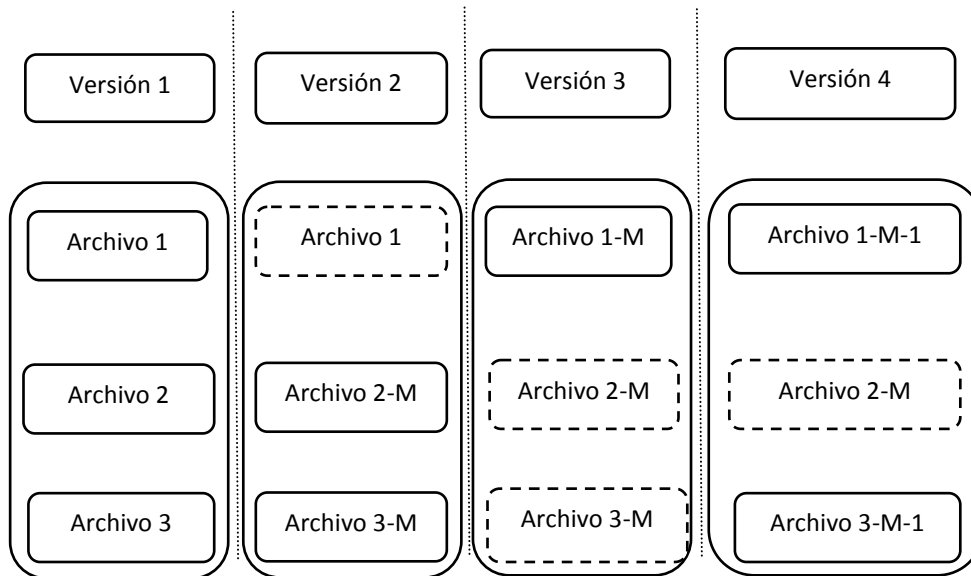


Figura 4.16 La figura muestra como Git almacena la información como instantáneas a lo largo del desarrollo y despliegue de las aplicaciones.

La manera en que Git modela sus datos la diferencian de otros sistemas.

Git modela sus datos más como un conjunto de instantáneas de un mini sistema de archivos, cada vez que se confirma un cambio o se guarda el estado de un proyecto en Git, básicamente lo que hace es una foto del aspecto de todos los archivos en ese momento, y guarda una referencia de esa instancia. En caso de que los archivos no se han modificado, Git no almacena el archivo nuevo, solo un enlace al archivo anterior idéntico al que ya tiene almacenado

En la Figura 4.16 se establece un proyecto con 4 archivos.

En la Versión 1, la primera versión de archivos es la de despliegue al usuario. Git toma la primera imagen de los archivos, guarda la referencia de todos los archivos en su primera versión.

En la Versión 2, los desarrolladores realizan un cambio en el Archivo 1. Git controla la versión de todo el proyecto a través de la toma de una imagen de todos los Archivos 1-2-M y 3-M y se guarda una referencia de este bloque de archivos.

Para el Archivo 1 que no fue modificado se guarda el enlace del Archivo 1 original.

En la Versión 3, Git toma una nueva imagen del conjunto de los Archivos 1M-2M-3M, con el nuevo Archivo 1-M modificados por los desarrolladores. En vista que los archivos 2-M y 3-M no fueron modificados se almacena el link de los archivos Archivo 2-M y Archivo 3-M anteriores.

En la Versión 4, Git toma una nueva imagen del conjunto de archivos de los cuales fueron modificados los Archivos 1-M de la Versión 3 y se convierte en el Archivo 1-M-1 en la Versión 4, y el Archivo 3-M de la Versión 3 pasa a ser el Archivo 3-M-1 en la Versión 4. Para el Archivo 2-M de la Versión 4, se almacena el enlace del Archivo 2-M de la Versión 3.

Las versiones de programas fuentes de Git son susceptibles de ser almacenadas y utilizadas en repositorios remotos (Ver Figura 4.17). Los repositorios remotos son versiones de programas fuentes que pueden estar en los servidores del Data Center distintos al computador local en el cual trabaja el desarrollador. De igual manera es

posible que el repositorio remoto se encuentre en la nube como lo hace Git Hub como se visualiza en la Figura 4.17. Git permite el manejo distribuido de varios repositorios remotos.

GitHub¹² es una plataforma de software colaborativo que sirve como repositorio para alojar programas o proyectos que son administrados con Git.

El almacenamiento de GitHub es en almacenamiento con acceso web público. El usuario crea su cuenta y se crea un link público en el cual se trabaja en el ambiente de GitHub con acceso desde cualquier sitio con acceso a internet.

El repositorio remoto que ofrece GitHub contiene todas las características del repositorio de Git en local, es decir contiene el historial de modificaciones que cada uno de los programas fuentes ha tenido desde el inicio de su versionamiento (Van, Rensen, Saboerali, & Teng, 2016). En la Figura 4.17 se establece un flujo de actividades desde que el desarrollador almacena su versión de la aplicación en el repositorio local que puede almacenarse en GitHub (Push) o ser recuperado de este a local (Pull).

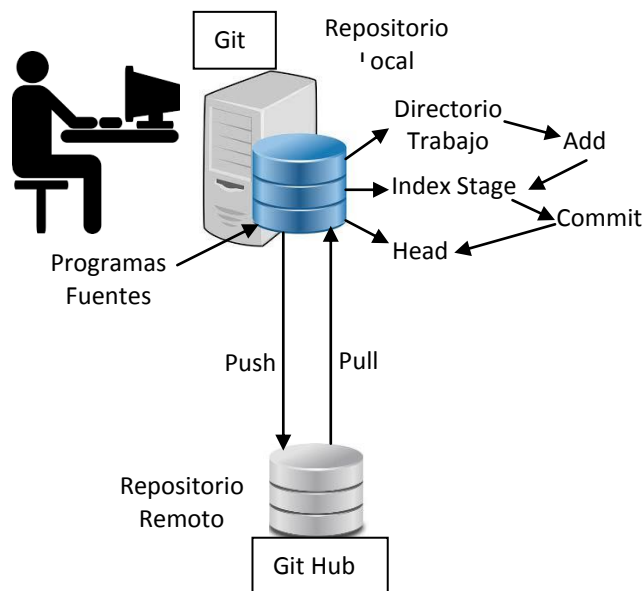


Figura 4.17 Secuencia de procesos de Git y GitHub

En la figura anterior se visualiza que al utilizar Git la versión que se encuentra en el directorio de trabajo, una vez modificada se transfiere al área de stage o index. Al hacer Commit, se genera un nuevo objeto luego del Stage producto del Commit, y se toma como base el nuevo producto.

4.7.3. Herramientas de Configuración

Las Herramientas de Configuración permiten automatizar y controlar tareas directas de los componentes del Data Center:

- Control de Inicio y Apagado de Servidores
- Ejecutar oportuna y correctamente los chek list diarios de revisión y control
- Instalación de sistemas operativos con recetas predeterminadas de acuerdo a las características de uso de ese momento.

¹² <https://github.com/>

- Ejecución en forma automática de Respaldos y Restore.
- Configuración de tareas de control de sistemas operativos, bases de datos, sistema de comunicaciones.
- Configuración de procesos batch del sistema o por usuario.
- Configuración en forma automática la creación de máquinas virtuales de servidores estratégicos o normales de trabajo, sea como respaldo o alta disponibilidad.

La información de configuración reside en un repositorio de código fuente. Software como Chef y Puppet, permiten a los desarrolladores tratar la configuración de los servidores que ejecutarán el software de aplicación como código (Robinson, 2016).

Chef permite automatizar la configuración de los componentes de la Infraestructura de un Data Center. Se define como un framework de automatización de infraestructura de sistemas (Chef, 2016).

Permite configurar, desplegar aplicaciones en cualquier ambiente físico o virtual o en la nube. El despliegue de software se refiere a todas las actividades que permiten que las aplicaciones estén disponibles para uso del usuario (Carzaniga et al., 1998).

Chef permite la instalación y configuración de pruebas similar al ambiente de producción.

Como su nombre lo indica, la herramienta pretende seguir un conjunto de pasos o recetas con el propósito de presentar un producto final ya listo para trabajar y/o probar (Chef, 2016).

Chef es flexible en su presentación y uso:

- a) Chef Server, se puede instalar la versión que funciona como repositorio de configuraciones necesarias para uso de instalaciones en los diferentes ambientes o nodos.
- b) Chef Solo, se puede instalar en un nodo en forma local lo cual permite tener todas las configuraciones e información de las recetas necesarias para configurar el nodo en el disco duro de la máquina nodo.

Cada uno de los procesos necesarios para instalar un determinado software o configurar una herramienta sigue determinadas secuencias, que son programadas en la herramienta DevOps Chef como definiciones reutilizables llamadas Cookbooks.

Los Cookbooks según (Chef, 2016), son archivos preparados para ejecutar comandos de acuerdo a la necesidad de instalación o configuración de software.

Es flexible en su presentación y uso. Los archivos son escritos en lenguaje Ruby y la preparación de los archivos se los realiza en secuencia de comandos denominados recetas.

Chef permite modelar la evolución de nuestra infraestructura y aplicaciones como si fueran código, permite describir y automatizar los procesos e infraestructura, la consecuencia es que la infraestructura se vuelve: Versionable, Testable, Replicable (Rodríguez et al., 2013).

Versionable, ya que permite el control de cada versión de software que se haya configurado, de esta manera en caso de tener varios servidores en los cuales se va instalar una determinada versión de software se lo hará en forma coordinada y estándar en cada uno de ellos.

Testeable, lo cual ayuda a realizar las pruebas necesarias antes de poner en producción cada una de las versiones del software, así como de comandos simples que se necesitan aplicar en un proceso batch.

Replicable, que permite instalar un mismo software en cada uno de los servidores lo cual permite contar con un inventario de las aplicaciones susceptibles de ser reutilizadas en otros servidores o áreas de TIC, con el fin de evitar el desperdicio de recursos y esfuerzos.

4.7.4. Herramientas de Gestión de Virtualización

Las Herramientas de Gestión de Virtualización ayudan a establecer ambientes virtualizados para distintos usos.

Vagrant¹³ es una herramienta que permite crear, configurar y reproducir entornos de desarrollo virtualizados en diferentes ambientes (Ver Figura 4.18).

Los entornos de desarrollo virtualizados ayudan estandarizar los ambientes de los programadores.

Vagrant permite automatizar los entornos de desarrollo virtualizados, de tal manera que los programadores pueden desplegar las máquinas virtuales con el ambiente que se ha definido como estándar de programación en sus máquinas locales.

En el ambiente del equipo de desarrollo se puede encontrar software que en algunos ordenadores funciona y en otro no, con Vagrant se puede trabajar bajo una misma copia, configurado en un mismo ambiente de desarrollo con sus dependencias y librerías establecidas como estándar.

Vagrant permite crear ambientes virtualizados similares a los de producción con los que interactúan los usuarios. La pruebas de las aplicaciones en ambientes virtualizados similares a los reales, ayudan a establecer las verdaderas capacidades del software desarrollado y permiten minimizar los errores que se puedan presentar al realizar el despliegue definitivo de las aplicaciones.

De esta manera se cumple con uno de los principales objetivos de Vagrant que es el de aproximar los entornos de desarrollo con los de producción, permitiendo a los desarrolladores tener a su disposición máquinas virtuales pre-configuradas de acuerdo a los convenciones entre las áreas de desarrollo y administración de sistemas.

Vagrant ayuda a resolver algunos problemas que se presentan en el desarrollo de software (Álvarez, 2015) (Van et al., 2016):

- A pesar que los programadores tienen en su máquina local los ambientes de desarrollo estandarizados, cada uno de ellos puede instalar en un momento determinado una librería o software con diferente versión para pruebas y no tuvo la precaución de comunicar al equipo de desarrollo y publica en producción con dichos cambios, lo cual puede producir problemas al momento de implementar las otras aplicaciones de los otros programadores.
- La aplicación puede funcionar en local y no en el servidor de aplicaciones.

Vagrant permite generar máquinas virtuales con especificaciones ligadas a las buenas prácticas. Es decir se puede diseñar cajas con distinto tipo de software (Ver Figura 4.18).

- Distintas versiones de sistema operativo Linux, Windows

¹³ <https://www.vagrantup.com>

- Cajas de Software para desarrollo con su manejador de base de datos, es decir front-end y back-end con sus respectivas dependencias.
Como ejemplo se puede estructurar cajas con entornos de desarrollo que se denominan LAMP, Linux-Apache-MySql-PHP.
- Estructurar cajas con software específico para aplicación de buenas prácticas de servicios, por ejemplo ITIL. La estructura se la realiza en base a los conceptos de servicios de Configuración, Versionamiento, Disponibilidad.
Puppet, Chef, ayudan a automatizar la configuración e instalación de software de acuerdo a las necesidades de cajas que se desea tener en disponibilidad para desarrolladores y operadores.
- Vagrant maneja un ambiente de scripts y se lo puede administrar aplicando prácticas de disponibilidad. Al manejar en forma estructurada la configuración de máquinas virtuales, se puede compartirlas de acuerdo a la necesidad de cada proyecto, optimizando tiempo y recursos de almacenamiento.

La arquitectura de Vagrant es la siguiente:

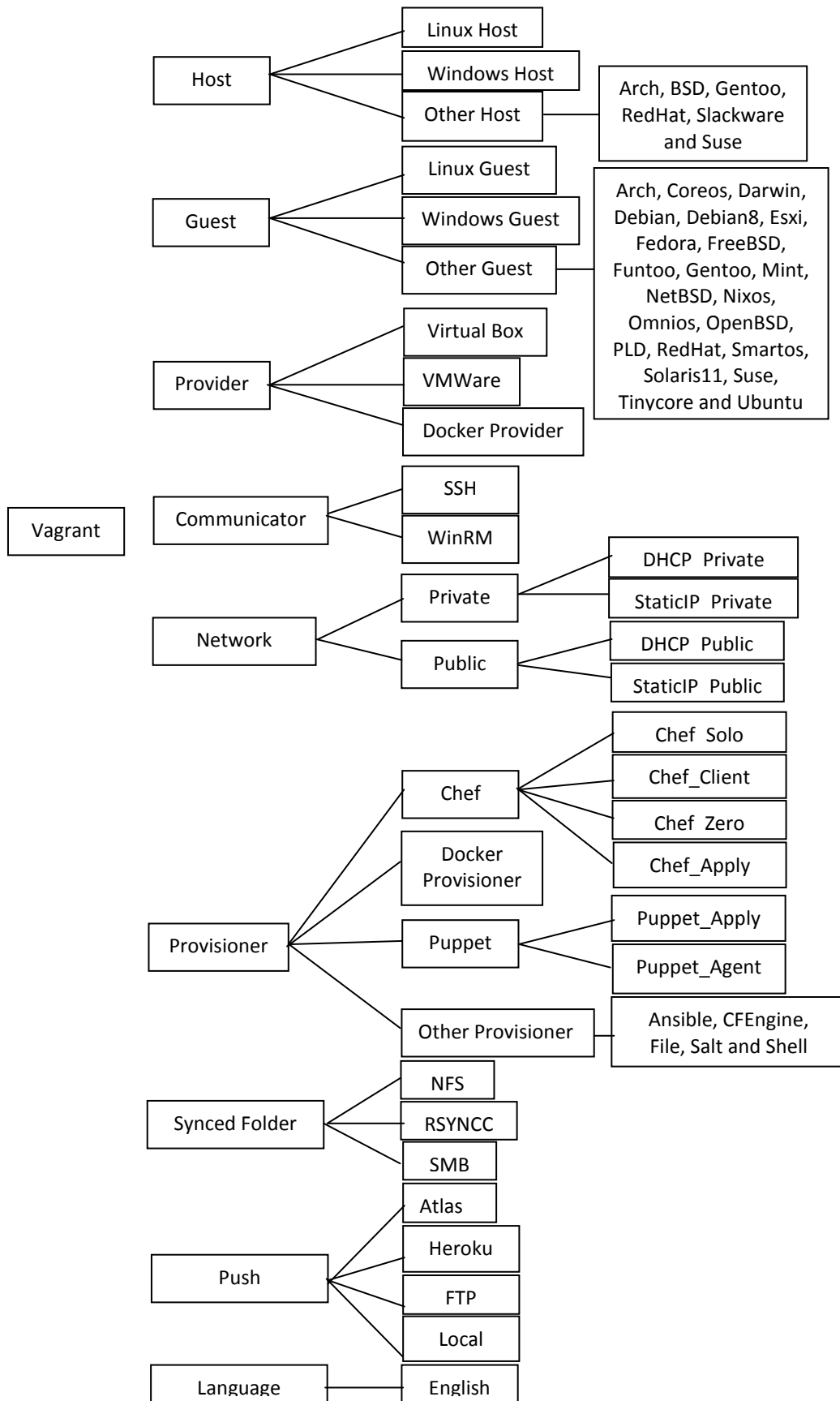


Figura 4.18 Estructura de Vagrant¹⁴

¹⁴ Adaptado de (Van et al., 2016)

4.7.5. Herramientas de Monitoreo

La Gestión de Monitoreo tiene por objetivo controlar la correcta ejecución de las tareas y servicios programados para una función específica. Uno de las herramientas destacadas es Nagios (Hernantes, Gallardo, & Serrano, 2015).

Nagios permite monitoreo de redes y componentes de software de servidores.

Permite automatizar las tareas a monitorear, actividades que son realizadas por los operadores en algunos casos en forma manual.

Nagios maneja un nivel de control adecuado para las características de monitoreo para aplicaciones comerciales.

Las características de configuración y sobre todo las propiedades de rendimiento que tiene Nagios, permiten un trabajo de monitoreo en espacios cortos de tiempo.

Una de las propiedades es la inmediatez.

A la Inmediatez se lo va a tratar como un tiempo delta t (Δt).

El tiempo delta t (Δt) es un rango de tiempo que permite ejecutar una secuencia de código que es configurado en los componentes que dan servicio a una aplicación, para detectar fallos y restaurarlos en forma temprana antes que se conviertan en averías.

Para lograr monitoreo que reporte información que ayude a la toma de decisiones es necesario realizar una planificación adecuada de los componentes que se va a monitorizar y relacionarlos con las aplicaciones que tienen relación dichos componentes. Esta es la clave del monitoreo. Sin una definición hacia donde se quiere llevar al análisis de la información resultante del monitoreo, no es posible configurar los atributos adecuados, ya que no hay un fin específico.

La toma de decisiones con la información que resulta del log de cada configuración de los componentes, debe estar acompañada de la gestión que realice el personal de administración de sistemas para lograr un control adecuado de disponibilidad de los componentes que brindan servicio a las aplicaciones que usan los usuarios.

La información que se reporta en los logs de monitoreo se puede automatizar en base a parámetros definidos y lograr módulos de acuerdo al área de ti que le corresponda revisar de acuerdo a su gestión.

La información también puede ser revisada en forma manual aunque resulte tedioso si no se cuenta con logs ordenados por tipos de información automatizados.

No tiene valor agregado almacenar información de log de monitoreo sin contar con recursos como personal que realice gestión sobre la información.

Debido a la gran cantidad de y variedad de componentes y servicios con los que cuenta el Data Center, es necesario conocer los componentes a monitorear, debido a que el log puede resultar demasiado engorroso analizarlo .

Entre los componentes que se puede monitorear con Nagios están:

- Monitoreo de servicios de sistemas informáticos
- Monitoreo de bases de datos relacionados con sistemas informáticos
- Monitoreo de sistemas operativos relacionados con sistemas informáticos
- Monitoreo de librerías relacionadas con sistemas informáticos
- Monitoreo de servicios de red relacionados con sistemas informáticos como:
 - Servicios de red (SMTP, POP3, HTTP, NTP, ICMP, SNMP)

- Monitorización de los recursos de un host relacionados con sistemas informáticos como:

- Carga del procesador, uso de los discos, logs del sistema en varios sistemas operativos, Linux, windows
- Diseño simple de plugins, que permiten a los usuarios desarrollar sus propios chequeos de servicios dependiendo de sus necesidades, usando algunas como:
 - Bash, C++, Perl, Ruby, Python, PHP, C#, Java
- Chequeo de servicios con fallos relacionados con los sistemas informáticos.
- Posibilidad de definir la jerarquía de la red, permitiendo distinguir entre host caídos y host inaccesibles.
- Notificaciones a los responsables cuando ocurren problemas en algún servicio. De igual manera enviar mensajes cuando son resueltos vía:
 - Email, pager, Jabber, SMS.
- Control de Servicios de Sistemas Operativos
 - Espacio de Filesystems
Se pueden presentar varios problemas en los Ambientes derivados de UNIX cuando un filesystem del sistema se llena, uno de ellos es que el servidor deja de operar y solo ingresando en modo de administración se puede vaciarlo, sin embargo no todas las versiones de sistemas operativos lo permiten.
En los servidores derivados de Windows no sucede lo mismo.
 - Uso de Memoria
 - Rendimiento de Procesador
Esta tarea es muy importante debido a que muchas veces problemas de mala configuración de las capacidades de un procesador con respecto al volumen de tareas y datos a ser manejados no son visualizadas al momento de adquirir un equipo. El monitoreo permitirá administrar y saber con pleno conocimiento un adecuado uso de los recursos.
 - Tarjetas de Red
- Control de Servicios de Bases de Datos
 - Espacio en Data Files
 - Listener activos
 - Usuarios activos
- Control de Servicios de Comunicaciones
 - Switch Core
 - Router Departamental
- Control de Servicios de UPS
- Control de Servicios de Aire Acondicionado
- Control y Monitoreo de Accesos Físicos al área de servidores
- Monitoreo de Seguridades Lógicas

La gran variedad de opciones de monitoreo puede dar opción a monitorear componentes que no entregan valor agregado a los servicios relacionados en forma directa con las aplicaciones usadas por los usuarios.

4.7.6. Protocolo de Monitoreo en la barrera de la tolerancia a Fallos

El siguiente protocolo se compone de acciones a seguir para lograr un monitoreo efectivo de las aplicaciones que se encuentran en producción.

- Determinar el objetivo del monitoreo
- Definir propiedades de apoyo de buenas prácticas apegadas a las necesidades del negocio que pueden ser disponibilidad, continuidad
- Determinar la herramienta de monitoreo de acuerdo a sus propiedades
- Determinar el rango de tiempo del monitoreo en la barrera de la tolerancia de fallos
- Definir los componentes de software macro relacionados con la aplicación a monitorear
- Detallar los componentes hasta un nivel atómico (que ya no tiene subcomponentes)
- Aplicar el monitoreo sobre el componente definido
- Realizar pruebas de funcionamiento

El objetivo se plantea desde la óptica de las necesidades del negocio.

Como resultado del monitoreo se pueden tomar acciones que ayuden a establecer y guiar las operaciones para asegurar el funcionamiento de las aplicaciones.

El objetivo debe estar acompañado de criterios de monitoreo que deben ser escogidos para que aporten valor agregado a la organización. De esta manera se logra que el monitoreo de las aplicaciones sea efectivo y se evita que se lo haga sin mayor definición, lo cual conlleva a ocupar espacio de almacenamiento infructuoso, cuyo análisis de la información no es fácil realizarlo.

Se presentan algunos objetivos ejemplos relacionados con la óptica de las necesidades del negocio:

- Monitoreo de las aplicaciones que se encuentran en producción.
- Monitoreo de las aplicaciones en producción que inciden en la relación del usuario con la selección de ofertas de productos que se ofrecen al mercado.
- Monitoreo de las aplicaciones que manejan los atributos de una organización que le distinguen y le dan valor agregado y ventaja competitiva sobre la competencia.
- Monitoreo de las aplicaciones que manejan las actividades y recursos de la organización que ayudan a dar fluidez y efectividad a la entrega de productos y servicios ofertados en el ámbito interno y externo.

Las necesidades de monitoreo deben ser planteadas en forma conjunta entre ejecutivos de la organización y las TIC.

El monitoreo debe ser planteado desde la óptica organización-aplicativo-usuarios. Al optar por estos elementos se puede definir conceptos de apoyo como la disponibilidad que se encuentra ubicua en las aplicaciones usadas por el usuario.

La disponibilidad apoya el alineamiento corporativo con las TIC y permite establecer heurísticas que ayuden a resolver problemas que pueden ser insignificantes para las TIC, pero imprescindibles para la organización y los usuarios.

La determinación de la herramientas de acuerdo a la propiedad de apoyo, implica que debe tener características que cumplan con las necesidades de monitoreo del negocio planteadas. Se expone a continuación, cómo se plantea esta determinación:

- Considerar una herramienta de monitoreo con propiedad de la Inmediatez.
- Una de herramientas que cumple con esta propiedad es Nagios. Permite ubicar secuencias de instrucciones en espacios de tiempos mínimos, en componentes de software del Data Center, susceptibles de ser configurados. De esta manera se obtiene control en puntos ruptura o posible fallo.

Determinar el rango de tiempo del monitoreo, significa los espacios de tiempo necesarios para operar la detección y corrección del fallo.

Con el fin de analizar la aplicabilidad de esta propiedad, se presenta la siguiente implementación que toma como referencia los conceptos de la barrera de la tolerancia a fallos, expuestos en la Figura 4.19 (Luque & Rexachs, 2014).

Fallo: Defecto o imperfección física en hardware o software del sistema.

Error: Estado interno incorrecto del sistema, es consecuencia de un fallo puede causar Avería.

Avería: El servicio entregado por el sistema no es el especificado, Usuario: el sistema no funciona bien.

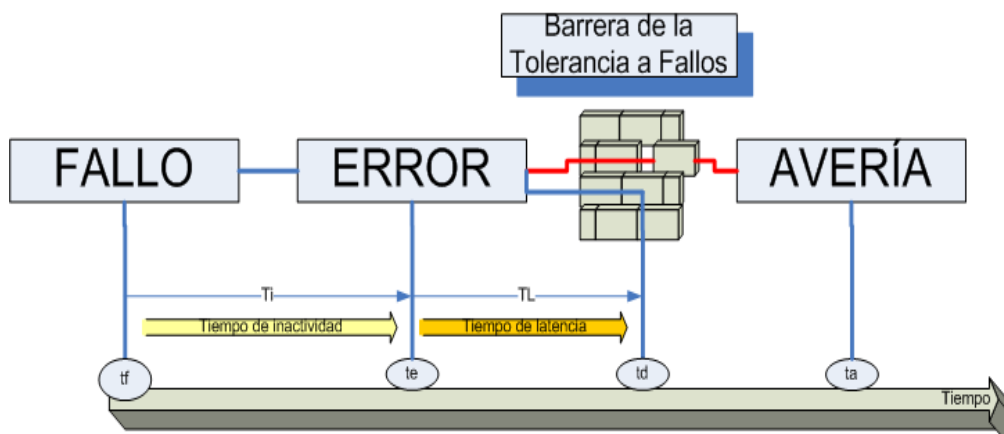


Figura 4.19 Tiempos de inactividad y latencia en tolerancia a fallos ¹⁵

Se abstrae que los atributos de monitoreo de Nagios se pueden manejar en esta barrera. El análisis de la figura anterior se lo enfoca desde el punto de vista de una aplicación de software:

- Una aplicación puede tener un fallo que no es detectado.
- El fallo se activa y es detectado el error, la aplicación puede seguir operando.
- Si el error no es corregido se puede convertir en avería.
- Al existir una avería la aplicación se detiene y deja de operar.

¹⁵ Figura tomada de (Luque & Rexachs, 2014)

- Nagios puede detectar el error en el espacio de tiempo que existe entre el error y avería. Así se aplica la propiedad de la Inmediatez.

Permite el seguimiento detallado de la interrupción y las causas que lo produjeron hasta llegar a su restauración, emitiendo las observaciones y correctivos adecuados.

La idea es alcanzar un monitoreo que permita niveles óptimos de disponibilidad de los servicios.

La disponibilidad de los servicios activos relacionados con los servicios de software, tienen relación directa con la Fiabilidad.

La Fiabilidad es la característica de los sistemas por la que se mide el tiempo de funcionamiento sin errores, y en caso de haberlos, controlarlos antes que se convierta en avería (Luque & Rexachs, 2014).

A fin de alcanzar una alta fiabilidad en una aplicación nos valemos de la inmediatez con su tiempo delta t (Δt), rango en el cual es posible realizar acciones desde la activación de un error hasta antes que se convierta en avería.

La explicación del párrafo anterior es así: si en el tiempo de error (t_e) existe control y monitoreo en una aplicación, que permita detectar el error, corregirlo y restaurarlo, no llegará al tiempo t_a , a ser avería.

Por lo tanto delta t (Δt) será mínimo y transparente para el servicio que se provee en ese momento permitiendo que la inactividad del servicio sea controlado.

La definición de los componentes de software macro relacionados con la aplicación a monitorear, consiste en definir si el control se lo va a realizar en componentes de software del sistema operativo o de base de datos.

Detallar los componentes hasta un nivel atómico (que ya no tiene subcomponentes), significa definir servicios que se puede denominar granulados (Sg), que ya no tienen más subcomponentes que dependan de su funcionamiento.

Aplicar el monitoreo sobre el componente definido, significa que ya se encontró con un nivel atómico de servicio. En este punto todavía se pueden tener varios servicios que pueden estar ocultos. Estos pueden aparecer con las diferentes pruebas de funcionamiento.

Como se visualiza en la Figura 4.20 en cada espacio de tiempo Δt se puede detallar y ubicar los servicios de cualquier tipo de componente que se analice.

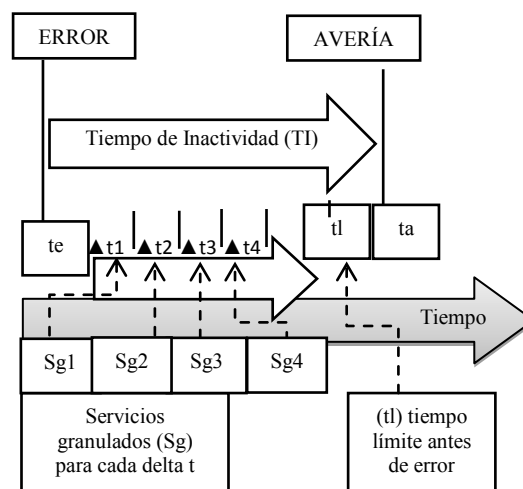


Figura 4.20 Tiempos permisibles para detección y control de fallos en forma automática

La explicación de los eventos que se presentan en la anterior es como sigue:

- En la detección de un error en un tiempo t_e se puede verificar qué servicio está caído para lo cual ha transcurrido un tiempo Δt_1 . Luego se envía el control al software que lo debe restaurar en un tiempo Δt_2 .
- En caso de existir error en tiempo t_e y no se pueda restaurar en el tiempo Δt_2 , se verifica si en Δt_3 o Δt_4 se lo puede activar. Si se llega al tiempo límite t_l , y no se lo ha restaurado, entonces hemos llegado al tiempo avería t_a . Esto significa que el servicio se detiene y dejar de operar.
- Un servicio granulado S_g , es un servicio que activa una aplicación. Si se detiene este servicio puede ocasionar que la aplicación deje de operar.
- En los tiempos Δt_1 , Δt_2 , Δt_3 y Δt_4 se pueden presentar varios servicios con errores, para esta situación se procede de la misma manera que el primer punto en forma paralela para Δt y S_g encontrado.
- La relación de servicios granulados y tiempos Δt es clave para llegar a un control automático con Nagios y sus módulos y paquetes individuales con funcionalidades para lograr un control y detección de errores en tiempos muy pequeños.

Si bien no se va a lograr tiempos de monitoreo a nivel de controles de sistemas en tiempo real¹⁶ debido a que el software de monitoreo DevOps Nagios está como capa del sistema operativo y no es parte del firmware del hardware, sin embargo se va conseguir tiempos de control y recuperación suficientes para lograr continuidad del servicio prestado.

¹⁶ Un sistema en tiempo real es un conjunto de componentes computacionales que interactúan en su entorno físico para responder peticiones de entradas o salidas dentro de un plazo determinado. Ejemplos de este tipo de sistemas son: control de vuelo de aviones, sistemas de control de distribución de energía eléctrica, entre otros. (De la Puente & Zamorano, 2002)(Stankovic, 1988)(Tejera, Alonso, & Miguel, 2011).

5. MODELO DE APOYO Y RELACIÓN ENTRE BUENAS PRÁCTICAS ITIL, MLS y MÉTRICAS CON DEVOPS

5.1. Estructura de Servicios ITIL y su relación con DevOps

La relación entre DevOps e ITIL abarca una nueva concepción del desarrollo y despliegue de aplicaciones. Se rompe el paradigma de trabajo en silos en las organizaciones (Ingall, 2003)(Paul M. Dooley, 2015).

Para llegar a una relación entre DevOps e ITIL se debe partir del alcance del Ciclo de Vida de DevOps.

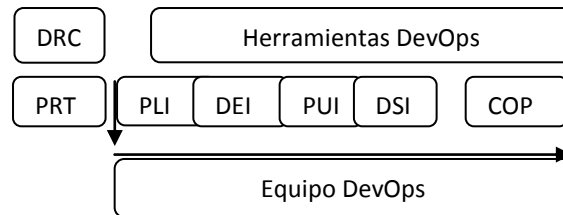


Figura 5.1 Alcance del Ciclo de Vida de DevOps

En la Figura 5.1 se visualiza que el equipo DevOps parte de la etapa de Planificación de la Iteración y continúa hasta la etapa de COP.

Cada etapa del Ciclo de Vida DevOps tiene su nivel de capacidad administrativa y operativa, por lo cual es necesario insertar las buenas prácticas de ITIL en cada fase con sus servicios para dar mayor estabilidad y efectividad al trabajo colaborativo de DevOps. Para que ITIL tenga un alineamiento adecuado en el Ciclo de vida de DevOps, se necesita detallar sus servicios y funciones.

5.2. Servicios de ITIL

Los servicios de ITIL son recursos que se entrega al usuario para ser utilizados en ámbitos del negocio. Deben ser enfocados al soporte de una actividad específica, a fin de lograr valor agregado.

Uno de los recursos de TI que requiere soporte de buenas prácticas es el desarrollo y despliegue de software. El software solo produce valor cuando está en producción y cumple con los requisitos solicitados por el usuario (Greer & Ruhe, 2003)(Akerle et al., 2014).

Es necesario identificar cada uno de los servicios de ITIL con el fin de verificar su aporte y dar madurez a cada fase del Ciclo de Vida DevOps (Ver Tabla 4-1), en función de la necesidad del negocio.

La identificación de las fases del Ciclo de Vida DevOps y la relación con cada servicio ITIL, debe ser sincronizada con las necesidades de requerimientos del dueño del producto. Esta concordancia, activa una alineación en todas las fases del desarrollo y despliegue de software y permite cumplir el avance de cada iteración, definida como prioritaria por el dueño del producto, de manera eficiente de acuerdo a sus requerimientos.

El ciclo de vida de los servicios de ITIL está definido en la Tabla 2-2, en ese contexto se va a establecer las relaciones respectivas con el ciclo de vida con DevOps.

5.2.1. Soporte del Servicio

El concepto de Soporte de Servicio se define como los medios para entregar valor a los clientes, facilitando sus tareas para obtener resultados, sin que ellos deban asumir los costos específicos ni los riesgos asociados (ITIL, 2015)(Ríos Huércano, 2014).

El Soporte del Servicio ayuda a coordinar la entrega adecuada de prestaciones que ofrece TI.

La Gestión de Soporte del Servicio se apoya en el área de Service Desk como primer contacto del usuario con personal de TI en el manejo de incidentes.

La Función Service Desk se define como un recurso que maneja un único punto de atención al usuario (ITIL, 2015)(Ríos Huércano, 2014).

Desde la Planificación de la Iteración hasta la etapa del Despliegue de la Iteración, el control de nuevas tareas lo mantiene el equipo DevOps. La Función Service Desk empieza una vez que ha sido desplegada la nueva aplicación al usuario, es decir empieza en la etapa de COP.

Al personal de Service Desk se lo comunica que existe una nueva aplicación que ha sido desplegada al usuario mediante una notificación automática, mediante mensaje, correo electrónico, etc.

Desde ese momento, los tiempos de respuesta de cada uno de los incidentes o problemas de las aplicaciones desplegadas deben ser mínimos por parte del servicio Service Desk.

Los Incidentes, son eventos que no forman parte de la operación normal del servicio, ocasionando reducción en la calidad de ese servicio (ITIL, 2015) (Raxl & Assaff, 2011).

Las aplicaciones desplegadas al usuario pueden tener errores en un momento determinado, dando lugar a un incidente.

El usuario reporta el incidente al personal designado como punto de primer contacto o Service Desk y se genera un ticket en forma automática para realizar el seguimiento respectivo.

Para lograr una correcta gestión de incidentes se debe seguir los siguientes pasos:

- Codificación y clasificación del incidente
- Ingreso del incidente
- Almacenamiento del incidente
- Resolución del fallo que ocasionó el incidente

Para dar solución al Incidente, se debe contar con personal capacitado en el entorno de las aplicaciones desplegadas al usuario.

En caso de no dar solución a los Incidentes presentados, se debe elevar la solución al equipo DevOps encargado de las aplicaciones desplegadas.

En caso de que un Incidente sea reiterativo se presenta un problema.

El Problema, se define como una condición identificada con frecuencia como el resultado de múltiples incidentes que presentan síntomas comunes (ITIL, 2015).

La adecuada gestión de problemas, incluye el análisis exhaustivo de las interrupciones que afectan al servicio para identificar causas y proponer soluciones que permitan evitar su repetición (ITIL, 2015).

Las aplicaciones que han sido desplegadas al usuario, pueden llegar a ser críticas para su labor y al no ser resuelto un incidente o un problema, se pone en duda la labor del Equipo DevOps.

Para evitar esta situación, el manejo de resolución de problemas se lo debe tratar con una planificación adecuada que incluya protocolos, procedimientos, actividades, y sobre todo, labor colaborativa entre personal de Service Desk, equipo DevOps y usuario, a fin de efectivizar la resolución del problema en tiempos cortos.

Esto conlleva a manejos estratégicos de análisis y resolución de problemas por parte del Equipo DevOps, que ayude a identificar cual es la causa-raíz del problema y en qué etapa de pruebas se pudo haber filtrado una condición de excepción que produzca el problema.

La Gestión de Cambios se activa una vez que han sido desplegadas las aplicaciones al usuario. Consiste en la evaluación y planificación del proceso de cambio para asegurar que, si éste se lleva a cabo, se haga de la forma más eficiente, siguiendo los procedimientos establecidos y asegurando en todo momento la calidad y continuidad del servicio TI (ITIL, 2015)(Ríos Huércano, 2014).

Para el equipo DevOps los cambios a ser realizados en las aplicaciones desplegadas deben ser mínimos. La coordinación de tareas en las etapas de PLI, DEI y PUI realizadas por el equipo DevOps, garantiza mínimos cambios luego de ser entregadas y desplegadas las versiones definitivas al usuario.

La gestión de los cambios en las aplicaciones que se encuentran en producción, deben ser tratadas en contexto de la metodología que se utilizó, con el fin de analizar el punto en el cual se originó el fallo que no pudo ser detectado al iniciar el desarrollo de la aplicación.

En caso de que el cambio necesite ser retornado a la etapa de Planificación de Priorización de Tareas, se debe empezar todo el ciclo de desarrollo DevOps para dar viabilidad a los cambios solicitados.

La gestión de cambios debe garantizar como objetivo primario la realización de cambios de procesos que tengan valor agregado, y con garantía que exista un mínimo de interrupciones en las aplicaciones que se despliegan al usuario. (ITIL, 2015).

La solicitud de los cambios a realizar en las aplicaciones desplegadas al usuario debe ser del mismo nivel y formalidad con el que se dio la petición inicial de planificación del proyecto automatización de software.

En la gestión de cambios, la PLI por parte del equipo DevOps debe ser consistente en las funciones asignadas, desarrolladores deben construir sus aplicaciones y operadores configurar sus servicios de administración de sistemas, y su labor en conjunto sigue siendo colaborativa de tareas, ideas, y configuración de herramientas.

Las funciones de la Gestión de Configuración permiten llevar el control de todos los elementos de la infraestructura de TI que van a ser utilizados por las aplicaciones que le son útiles al usuario (ITIL, 2015).

La Gestión de Configuración permite que todos los elementos de software y hardware sean gestionados y almacenados en la Base de Datos de Administración de Configuración (CMDB) (ITIL, 2015).

El almacenamiento de componentes de software almacenados en la base de datos de configuración proporciona información sobre el software utilizado, tanto en desarrollo como en el despliegue de las aplicaciones.

La Gestión de Configuración coordina que equipo DevOps realice, la configuración de herramientas computacionales desde la PLI en dos sub-etapas:

- Sub-etapa de configuración de software, librerías a ser utilizadas por el equipo desarrollador (Dev).

- Sub-etapa de pre-configuración de herramientas en servidores por parte del equipo de operadores (Ops).

En las dos sub-etapas los desarrolladores y operadores tienen responsabilidad propia de sus herramientas, no obstante su ambiente DevOps les permite un trabajo colaborativo para configurar software y hardware compatible con la aplicación final que se va a entregar al usuario. Este proceso minimiza los fallos en la etapa de despliegue.

La colaboración de desarrolladores y operadores desde la etapa de PLI permite dar ideas innovadoras de tecnología de punta con software y hardware de acuerdo a las tareas a ser automatizadas.

En este contexto la Gestión de Configuración puede interactuar con las Gestiones de Incidentes, Problemas, Cambios y Versiones de manera que estas puedan resolver más eficientemente las incidencias, encontrar rápidamente la causa de los problemas, realizar los cambios necesarios para su resolución y mantener actualizada en todo momento la CMDB.

Las configuraciones son dinámicas y cambiantes debido al avance de necesidades diarias de los usuarios. Para ello se vuelve indispensable contar con software especializado que permita realizar seguimientos y controles que ayuden a una correcta administración de configuraciones.

La Gestión de Versiones inicia el momento que la iteración es desplegada al usuario, con el control automático de la versión de los componentes de software y hardware relacionados con la versión de la aplicación que se encuentra en producción.

La Gestión de Versiones es la encargada de la implementación y control de calidad de todo el software y hardware instalado en el entorno de producción.

El equipo DevOps debe soportarse en herramientas que ayuden a un control detallado de las versiones de todos los componentes relacionados con la aplicación.

Es clave la identificación de los componentes de software y hardware que se encuentran directamente relacionados con las aplicaciones que han sido desplegadas al usuario.

La petición de nuevos cambios a las aplicaciones genera nuevas versiones de aplicaciones y a ellas se encuentran relacionadas versiones de software y hardware de los distintos componentes relacionados.

La Gestión de Versiones asegura que toda la información relativa a las nuevas versiones se integre adecuadamente en la CMDB de forma que ésta se halle correctamente actualizada y ofrezca una imagen real de la configuración de la infraestructura TI.

Entre los componentes que debe coordinar y controlar la Gestión de Versiones están:

- Versiones de software de servidores
- Software de estaciones de trabajo
- Control de Licencias
- Versiones de software de aplicaciones
- Versiones de software de virtualización de servidores
- Versiones de software de software de estaciones de trabajo
- Versiones de librerías de sistema operativo
- Versiones de software de bases de datos
- Versiones de software de intranet
- Versiones de servidores y aplicaciones web

5.2.2. Entrega del Servicio

La Entrega del Servicio permite establecer y administrar los servicios de TI, con los niveles de servicio requeridos por el negocio (Hrabinski, 2009) (Weill, Subramani, & Broadbent, 2002).

La entrega del servicio de aplicaciones empieza desde la DRC a ser automatizados hasta llegar a la etapa de COP. Cada una de las etapas del desarrollo de software ofrece un servicio a la organización.

Se detalla en la Tabla 5-1 los servicios que entrega TI en las etapas de desarrollo y despliegue de aplicaciones.

Tabla 5-1 Servicios en el ciclo de vida de DevOps

Servicio
Servicio de definición de requerimientos a ser automatizados
Servicio de planificación de priorización Tareas
Servicio de planificación de iteración
Servicio de desarrollo de iteración
Servicio de pruebas unitarias
Servicio y despliegue de iteración
Servicio de control en producción

La Gestión de la Capacidad permite asegurar que la capacidad de los servicios de infraestructura de TI sea el adecuado para cubrir las necesidades del negocio en forma oportuna.

La Gestión de la Capacidad comprende procesos de capacidad del negocio, de los servicios y de los recursos.

La Gestión de la capacidad del negocio, es responsable de la planificación de las futuras necesidades del negocio.

DevOps no interviene en la gestión de la capacidad del negocio, ya que es una planificación macro de todas las necesidades futuras de TI.

La gestión de la capacidad de los servicios, permite asegurar que los servicios de infraestructura mantengan los acuerdos de nivel de servicio acordados.

La gestión de la capacidad de los servicios es clave en DevOps. Parte desde la PLI en acuerdo con el dueño del producto. Se coordina tareas, recursos, tiempos cortos para cada iteración.

La gestión de la capacidad de los recursos, se encarga de gestionar los recursos de cada componente de la infraestructura que se utiliza para dar servicio a las aplicaciones del usuario.

El alineamiento entre la gestión de la capacidad de los recursos y DevOps es directo. En cada iteración que se va desarrollando se debe tener configurado los componentes en capacidad de infraestructura virtual y física.

La gestión de la capacidad de los servicios y la capacidad de los recursos garantizan que el equipo DevOps cuente con la capacidad de infraestructura necesaria para cada iteración desplegada como para todo el proyecto de software.

La gestión de costos de TI, es un proceso de ITIL que tiene el equipo DevOps para coordinar acciones de alineamiento adecuado entre todos los componentes a ser

utilizados en el desarrollo y despliegue de aplicaciones con el fin de que los costos sean razonables por cada iteración.

Cada etapa que ofrece un servicio, requiere ser alineado con buenas prácticas que efectivicen el desarrollo y despliegue de las aplicaciones.

Uno de los objetivos de ITIL es aumentar la confiabilidad de la infraestructura y los servicios que se proveen al usuario (ITIL, 2015)(Gary & Jimmy, 2008). Las buenas prácticas ITIL permiten dar pautas para coordinar las acciones que ayuden a tener un control sobre los componentes que dan servicios para el desarrollo y despliegue de las aplicaciones.

Las responsabilidades que tiene la Gestión de la Disponibilidad de ITIL (ITIL, 2015)(Gary & Jimmy, 2008)(Hrabinski, 2009) permiten:

- Garantizar el nivel de disponibilidad de servicios establecido para los servicios de TI.
- Monitorizar la disponibilidad de los sistemas TI.

Para garantizar el nivel disponibilidad de los servicios establecidos para las prestaciones que se dan en el ciclo de vida con DevOps, cada fase debe ser estructurada en labores colaborativas de ciclos de tiempo corto, mismos que se analizan en la planificación de la iteración.

La etapa de PLI por sí sola no garantiza disponibilidad del servicio que se genera en cada etapa. Se deben identificar los servicios que brindan los componentes y subcomponentes de software e infraestructura relacionados hasta el despliegue de la aplicación.

Cada componente que brinda un servicio para que la aplicación del usuario esté activa, tiene espacios de tiempo y formas de ser controlado, lo cual se estructura en la etapa de DEI hasta el DSI y COP.

En la etapa de COP es necesario apoyarse en herramientas de monitoreo y restauración con operaciones automáticas sin necesidad que el personal de TI actúe y únicamente conozca los fallos producidos a través de mensajes o correos electrónicos enviados por la herramienta usada para el restablecimiento del servicio.

La Gestión de Operaciones ayuda a establecer mecanismos para que el personal de operaciones lleve a cabo un adecuado control de administración de los sistemas (ITIL, 2015)(Hrabinski, 2009).

Entre sus responsabilidades se encuentra la gestión de:

- Despliegue de máquinas virtuales para desarrollo o pruebas de aplicaciones
- Despliegue automático de aplicaciones
- Definición y configuración de script para ejecución de aplicaciones en batch
- Instalación y configuración de sistemas operativos, bases de datos, frameworks a ser utilizados por los desarrolladores.

La Automatización, es un componente estratégico de las actividades de soporte con ITIL. Los hitos en los que se desenvuelve la automatización están enfocados desde el inicio de la DRC hasta el COP.

La automatización de las tareas involucra actividades de desarrollo y operaciones en el ámbito que les compete. Desarrolladores en las aplicaciones y operadores insertando

código en las tareas de configuración y monitoreo de componentes de despliegue de aplicaciones.

La automatización de componentes para el despliegue de aplicaciones por parte de los operadores, va a servir para que toda la entrega de prestaciones a los distintos usuarios como el área de desarrollo se la realice en forma automática.

El despliegue de recursos como máquinas virtuales y todo el conjunto de software que se necesita para pruebas o producción, debe realizarse en forma automática, sin la dependencia del personal de operaciones.

La Gestión de Seguridad, se encarga de planificar las labores operativas, tácticas y estratégicas que brinden seguridad a todos los servicios relacionados con las aplicaciones de los usuarios (Hrabinski, 2009).

La Gestión de Seguridad operativa tiene estrecha colaboración con las labores del equipo DevOps, permite reducción del riesgo en todas las etapas del desarrollo y despliegue de la aplicación al usuario, como del seguimiento del producto final.

La planificación táctica coordina los elementos que brinden seguridad en todas las etapas del desarrollo y despliegue de aplicaciones.

La planificación estratégica gestiona la planificación de seguridad global de TI. DevOps es un subconjunto de esta planificación.

5.3. Puntos de Alineamiento Estratégico entre ITIL y DevOps

La estructura de alineamiento entre ITIL y DevOps tiene tres ambientes.

El primer ambiente en el que ITIL y DevOps no se alinean, corresponde a la planificación de los requerimientos del usuario (Ver Figura 5.2.1).

En el primer ambiente no se tienen definidos los Requerimientos a ser Automatizados.

Este punto es denominado zona oscura y representa el ambiente en el cual solo intervienen los desarrolladores y el dueño del producto (Hüttermann, 2012).

En la zona oscura no hay relación entre ITIL y DevOps.

ITIL si puede estar alineado con las etapas que intervienen en la zona oscura, pero al no intervenir el equipo no es posible relacionarlo con DevOps.

El segundo ambiente corresponde a la etapa de DRC, solo interviene ITIL (Ver Figura 5.2.2).

En el segundo ambiente todavía no se encuentra conformado el equipo DevOps, el Product Backlog o lista de requerimientos se encuentra en análisis y planificación tratando de dar un orden de prioridad a las tareas, así como los tiempos y responsables de cada una de ellas.

El tercer ambiente es el ambiente colaborativo entre ITIL y DevOps (Ver Figura 5.2.3).

En este ambiente, empieza la etapa de PLI, en donde ya se conocen las tareas prioritarias y por lo tanto la colaboración del equipo de operaciones se hace indispensable. Al ingresar operaciones se estructura el equipo DevOps, por lo tanto ya es posible relacionar con ITIL.

ITIL forma parte del alineamiento desde esta etapa(PLI) hasta el COP de las aplicaciones como se puede ver en la Figura 5.2.

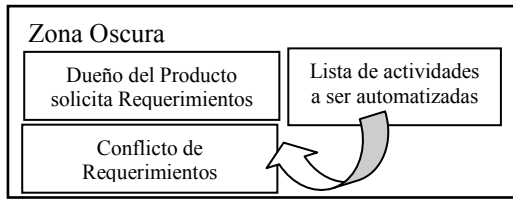


Figura 5.2.1 Primer ambiente: Zona oscura
No interviene ITIL y DevOps

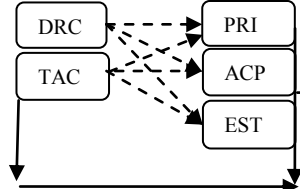


Figura 5.2.2 Segundo ambiente: Zona de influencia de ITIL, no interviene DevOps

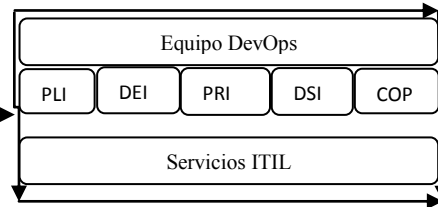


Figura 5.2.3 Tercer ambiente: Zona de apoyo de ITIL en todo el ambiente de desarrollo y despliegue con DevOps

Figura 5.2 Ambientes de influencia de DevOps e ITIL

En la Figura 5.3 se puede verificar el tercer ambiente que son los puntos congruentes de apoyo entre DevOps e ITIL:

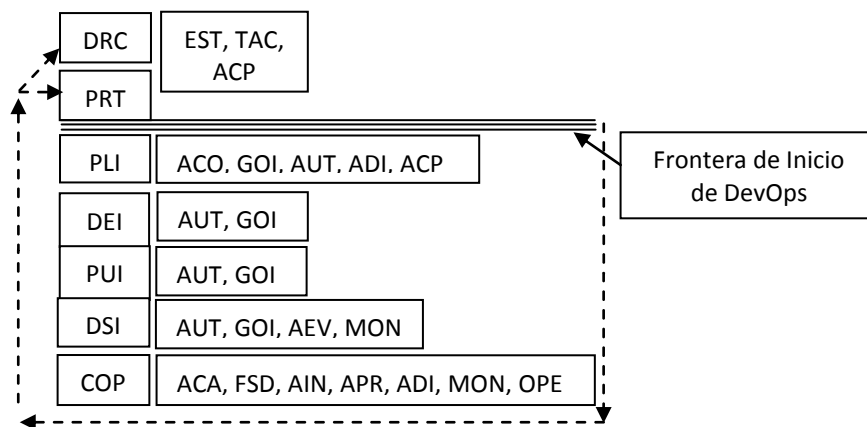


Figura 5.3 Relación entre componentes DevOps e ITIL

A continuación se presenta una breve explicación de la figura anterior:

EST, TAC, ACP, son servicios de ITIL alineados a la planificación general del proyecto de software. Se estructuran las estrategias para lograr la definición de requerimientos y los aspectos tácticos que ayuden a clarificar las necesidades a ser automatizadas. Se especifican las tareas de acuerdo a su prioridad, y con base en ellas se planifica la capacidad de los recursos donde se van a alojar los componentes de los servicios relacionados con las aplicaciones.

ACO, GOI, AUT, ADI, ACP, son servicios ITIL que guían el proceso de desarrollo y despliegue de aplicaciones, desde la planificación de la iteración hasta el control de

producción, de acuerdo al servicio de buenas prácticas que le corresponde en cada etapa del ciclo de vida.

AEV, MON, son servicios de ITIL que interactúan con el control de versiones y monitoreo al momento de ser desplegada una nueva iteración.

ACA, FSD, AIN, APR, son servicios de ITIL que dirigen sus buenas prácticas una vez que ha sido desplegada una iteración al usuario. ACO, GOI, AUT, ADI, ACP, AEV, MON, continúan alineados con la labor del equipo DevOps.

Las fases de desarrollo y despliegue de aplicaciones y los componentes de ITIL van de la mano de acuerdo a la necesidad de apoyo en cada tarea que se va desarrollando.

El alineamiento de ITIL con DevOps permite aplicar buenas prácticas en cada etapa del ciclo de vida del desarrollo y despliegue de aplicaciones. Este engranaje resulta efectivo desde el inicio de la iteración y se puede precisar que en todos los componentes DevOps se pueden alinear con uno o más componentes de ITIL, ayudando a dar fortaleza a la entrega de aplicaciones.

5.4. Estructura de relación de principios de MLS con DevOps

Un Startup permite realizar variantes en diferentes niveles, basados en la incertidumbre. No es posible encontrar soluciones definitivas al enfrentar el desarrollo y despliegue de aplicaciones. Siempre se está evolucionando y aprendiendo conforme se arma un producto para el usuario.

En este contexto MLS ayuda con sus principios a establecer estructuras ligeras que permitan entregas inmediatas de productos para validación del usuario.

5.4.1. Relación de MLS y DevOps

Los principios anotados en la sección 2.3 dan su apoyo a todas las etapas del ciclo de vida DevOps.

Desde la etapa de PLI hasta la etapa de COP, el soporte de los principios MLS es estratégico ya que se produce un alineamiento directo entre los requerimientos del usuario con el equipo de TI.

El apoyo de la Metodología Lean Startup con sus principios al Ciclo de Vida DevOps, influye en la calidad del desarrollo de software.

Los Principios Crear, Aprender, Optimizar de Lean Startup, dan soporte a cada etapa del Ciclo de Vida DevOps, PLI, DEI, PUI, DSI, COP.

La creación de nuevos servicios empresariales implica organizar, optimizar los procesos administrativos, técnicos, operativos organizacionales y TI, que permitan dar valor agregado a las competencias distintivas de la Organización.

El principio crear rige en cada etapa del ciclo de vida DevOps. Se fundamenta en la creación de productos con valor agregado. No establece la automatización de tareas sin aporte de alineamiento con los objetivos empresariales.

El principio de aprender aporta al equipo DevOps con los siguientes elementos:

- Todos los miembros del equipo DevOps aprenden basados en el principio de trabajo colaborativo. El aporte de conocimiento técnico, de procesos administrativos, produce un efecto de eficacia hacia el producto que se está creando.
- El equipo DevOps debe aprender de los errores que se presentan en el desarrollo y despliegue de aplicaciones. Sirve para establecer controles

en base a la experiencia de fallos. Este aprendizaje en etapas tempranas del ciclo de vida DevOps ayuda a disminuir los errores en producción.

El principio de Producto Mínimo Viable (PMV) apoya al Ciclo de Vida DevOps en la búsqueda de iteraciones cortas que permitan descifrar la incertidumbre que ocasiona la falta de definición de requerimientos.

Cada etapa del ciclo de vida DevOps está basada en crear PMV. El desarrollo de aplicaciones con integración continua también se ve influenciado con el PMV, se va integrándolo de acuerdo a la secuencia de PMV construido por los desarrolladores y usuarios participantes.

A los operadores les resulta conveniente definir PMV ya que la configuración de componentes del despliegue de aplicaciones se vuelve menos dinámica.

Al construir cada iteración en corto tiempo se vislumbra más fácilmente los problemas que presenta el usuario al definir sus tareas a automatizar.

En la etapa de COP de las aplicaciones se activa el monitoreo por parte del equipo DevOps ya que mantienen su influencia verificando que los requerimientos sean óptimos y automatizados de acuerdo a las necesidades de los usuarios.

La etapa DevOps que entra en conflicto con MLS, es DEI.

En el desarrollo de la iteración se asume que la tarea a ser automatizada, así como los recursos a ser utilizados ya están definidos y por lo tanto no se permite el aporte de nuevas ideas. Por lo tanto la etapa de DI cierra todo tipo de intervención a nuevos aportes de innovación mientras se está desarrollando la iteración.

5.5. Aporte de Métricas a DevOps

Uno de los puntos congruentes entre DevOps y la MLS es el principio de la Métrica.

La Métrica se define como la medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado (IEEE, 1990).

Es necesario que los gestores de TI y del negocio puedan alcanzar acuerdos claros sobre la forma de realizar las mediciones de los objetivos de desempeño y conformidad de la organización. Esto permitirá que los procesos y actividades sean medidos en función de metas y objetivos dentro de la organización y conocer que áreas necesitan una atención especial (Chan, 1992).

Los atributos de las métricas se dan por acuerdo entre directivos y personal técnico, operativo, administrativo, en base a un valor que mide la gestión en un tiempo determinado.

Las métricas permiten a las áreas de desarrollo de software y operadores, determinar indicadores que miden su desempeño. Califican la eficacia de su labor en forma aislada para cada área.

El marco colaborativo en el que se desarrolla DevOps rompe el silo en el que cada área mide su labor de acuerdo al paradigma de rendimiento acostumbrado.

Las Métricas de DevOps ayudan a mantener el equilibrio entre la velocidad de desarrollo, el nivel de calidad del software y el costo de software.

La velocidad en el desarrollo de software puede generar disminución en la calidad y el desarrollo de software en tiempos óptimos con altos niveles de calidad puede producir aumento del costo.

DevOps promueve fundamentalmente: una cultura abierta, de colaboración y transparencia entre los equipos de desarrollo y operaciones, donde la responsabilidad combinada de los equipos predomine sobre la eficiencia y límites laborales individuales.

Con el fin de determinar los mejores indicadores de desempeño en base a DevOps, se determinan varios desafíos que deben enfrentar las organizaciones y los clientes en el despliegue de software (Bourne, 2013):

- Lentitud en despliegue del software solicitado
- Mala experiencia del usuario
- Alto costo
- Poca Previsibilidad
- Vulnerabilidad y Riesgos

Es muy probable que las empresas tengan que enfrentar uno o varios de estos desafíos. Por esto es recomendable que las organizaciones de TI adopten un nuevo enfoque para la definición de algunas métricas que sirvan de objetivo a alcanzar por parte de los equipos que inician la práctica de DevOps.

Casi todas las empresas se enfrentan a por lo menos tres o cuatro de estos desafíos, y muchas empresas se enfrentan a los cinco. No es de extrañar que muchos equipos de software no conozcan dónde ni cómo enfrentar sus problemas (Bourne, 2013). Las causas de estos problemas se reducen a cuatro elementos:

- Velocidad
- Calidad
- Productividad
- Seguridad

La Velocidad representa la tasa de entrega de nuevo software o de cambios realizados al software del usuario.

La Calidad permite medir si el software desarrollado o los cambios entregados satisfacen los requerimientos del usuario

La Productividad representa la capacidad y eficiencia de una organización para entregar nuevo o realizar cambios al software del usuario.

La Seguridad representa la medida de las vulnerabilidades que se pueden presentar al desarrollar o realizar cambios al software del usuario.

Este marco de métricas indica dónde está teniendo éxito y donde todavía se necesita mejorar (Bourne, 2013).

Gartner define nuevas perspectivas con métricas para el ambiente DevOps.

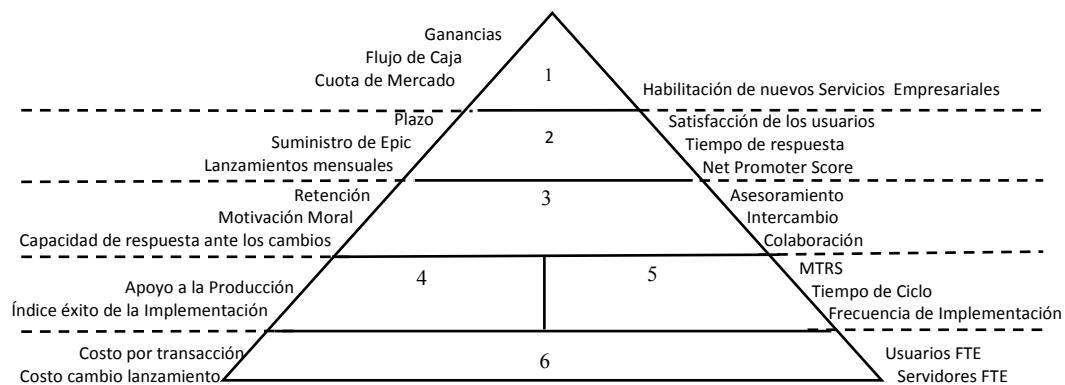
Gartner plantea 5 dominios para control de métricas con DevOps, que permiten tener una visión más clara de evaluación de avance de las etapas en el desarrollo y despliegue de aplicaciones.

1. El Negocio desde el nivel Empresarial
2. Dominio del Cliente
3. Dominio de la Organización
4. Análisis del Servicio
5. Administración de Sistemas

Estos 5 dominios son presentados en la Figura 5.4 como la Pirámide de Métricas de DevOps.

Cada uno de los dominios son expuestos con sus métricas de acuerdo a la característica empresarial sea de negocio o de apoyo.

El Alcance del Negocio proporciona servicios internos y externos a la organización.



1 Rendimiento Empresarial 2 Valor para los Clientes 3 Efectividad Organizativa 4 Calidad del Servicio 5 Velocidad del Servicio 6 Eficiencia Operativa

Figura 5.4 Pirámide de Métricas de DevOps (Fuente Gartner 2014)

La explicación de cada elemento de la Figura 5.4 es como se muestra a continuación.

Las métricas en el dominio del Rendimiento Empresarial, están enmarcadas bajo el contexto de su dimensión de conformidad.

Las métricas empresariales buscan medir variables que ayuden a definir nuevas inversiones, proyectos, estrategias que contribuyan a mejorar el rendimiento de las áreas involucradas con el fin de cumplir los objetivos empresariales. El portafolio de proyectos que maneja la estrategia de la organización incidirá en la creación y habilitación de nuevos servicios.

Una de estas variables es la habilitación de nuevos servicios empresariales. La estrategia que adopte la organización para realizar nuevos emprendimientos, está relacionada con la estrategia de TI para dar un soporte que contribuya a entregar información y recursos computacionales para que la organización tenga éxito.

Las Métricas en el Dominio del Cliente permiten realizar una evaluación del valor de los servicios prestados al cliente (Haight & Bandopadhyay, 2015).

Los Plazos, los Tiempos de Respuesta, los Lanzamientos Mensuales, dan la medida del funcionamiento de los procesos que lleva a cabo el equipo DevOps para dar respuesta a los requerimientos del usuario.

El requerimiento puede ser una aplicación nueva, una modificación o cambios requeridos, que implique la intervención del equipo y del ciclo de vida DevOps.

La intervención del usuario definirá la estrategia para planificar los plazos de las tareas prioritarias a ser automatizadas.

Los tiempos de respuesta son definidos en la etapa de PLI, que pueden ser optimizados de acuerdo a la estrategia que tenga el equipo DevOps para hacer entregas parciales de PMV.

Los lanzamientos mensuales de una iteración pueden ser cambiados a lanzamientos diarios, semanales con el fin de que el usuario interactúe en forma más temprana con los servicios entregados en base a los requerimientos solicitados y a fin de disminuir los fallos posibles, aumentando la efectividad del producto final.

El Suministro de Epic, permite analizar las historias de usuario o grandes requerimientos de los usuarios difíciles de estimar.

Las Épicas se relacionan con las historias de usuarios robustas que pueden ser ambiguas por su extensión. Con las Épicas se crean el Product Back Log.

Debido a que las Épicas son extensas no permiten dimensionar los trabajos para las tareas de los desarrolladores, por lo tanto deben ser divididas en Épicas más pequeñas (aplicación del PMV, que significa diseñar épicas funcionales tan pequeñas como sean posibles).

Net Promoter Score ayuda a evaluar el valor esperado mediante la medición de la fidelidad de los clientes. Evalúa el producto o servicio ofrecido al usuario mediante la verificación y validación de una encuesta en la cual se comprueba si el servicio ofertado puede ser recomendado a una tercera persona amigo o familiar.

La escala con la cual se realiza la encuesta ayuda a definir si el producto o servicio satisface las necesidades del usuario. Se la puede realizar mediante hojas formularios o página web preparada para el efecto.

El Dominio de la Organización mide la efectividad que tiene la organización para entregar productos con valor agregado al usuario/cliente.

El análisis de la Métrica de La Retención ayuda a identificar el impacto que produce el producto o servicio que se está entregando al cliente (Megias, 2013).

La Métrica de La Retención en el Dominio de la Organización marca dos enfoques:

- Enfoque desde el punto de vista de la organización cuyo producto o servicio está causando impacto en el cliente. De acuerdo al nivel de venta del producto se determina lo útil que es el producto para el usuario.
- Enfocado a DevOps implica un análisis que ayuda a determinar que los sistemas que están siendo usados por el usuario son verdaderamente atractivos que ayudan a causar impacto y dan valor agregado al producto que se está ofertando al usuario por parte de la organización.
Existen variables ayudan a comprender si los componentes informáticos con los cuales interactúa el cliente, son los factores determinantes en el impacto del producto al usuario:
 - Sistemas amigables con comprensión clara de lo que se está ofertando
 - Detalle técnico del producto ofertado
 - Costos y beneficios del producto

Estos enfoques se deben analizar muy detenidamente con el fin de no caer en la perspectiva de ofrecer un producto sin mayor impacto para el usuario, a pesar de la publicidad muy bien realizada y diseñada que se ofrece con los sistemas informáticos o páginas web.

La Motivación Moral como Métrica del Dominio de la Organización identifica el grado de satisfacción que tiene el equipo de trabajo al realizar su trabajo en un ambiente determinado.

No es fácil definir si algún miembro del equipo de trabajo se encuentra con la moral alta, media o baja. El trabajo en equipo puede ocultar el verdadero sentimiento de un miembro del equipo. La capacidad de trabajo de un miembro del equipo difiere del otro en un mismo estado de ánimo y viceversa, es decir no todos los miembros del equipo

van a tener un nivel de productividad efectivo así ellos tengan un alto grado de profesionalismo el desarrollar su trabajo.

La Capacidad de Respuesta ante los Cambios representa el grado de efectividad que tienen los componentes de recursos humanos, administrativos, técnicos ante las solicitudes de cambio de clientes o usuarios de los procesos que intervienen en la provisión de servicios ofertados por la organización.

La capacidad de respuesta ante los cambios también tiene que ver con el factor de la motivación moral que tenga cada uno de los miembros del equipo de trabajo.

El Asesoramiento como Métrica de DevOps permite definir las áreas que necesitan asesoramiento adecuado para la continuidad de sus operaciones y lograr calidad en el desarrollo de software.

Las Métricas de Intercambio y Compartición parten de sus conceptos para dar su aporte a DevOps (Estrella, 2011).

El Intercambio tiene como fin el ofrecer un bien que puede ser material, conocimiento, experiencia, que le interesa a otra persona y viceversa. El beneficio satisface a las dos personas.

La Compartición que es principio de DevOps, se puede entender que es la que reúne los rasgos semánticos de: repartir, distribuir algo en partes o participar en algo¹⁷.

Tanto el Intercambio como la Compartición tienen variables que participan en sus conceptos:

- Ideas
- Conocimiento
 - o Técnico
 - o Administrativo
 - o Procesos

El Intercambio y Compartición de Ideas, Conocimiento, es estratégico en todas las etapas del ciclo de vida de DevOps.

El Intercambio y la Compartición son conceptos claves que encajan en el manejo adecuado de un trabajo en equipo de desarrollo y despliegue de software.

Las Métricas de Análisis del Servicio, analizan la velocidad y calidad del servicio que se ofrece al usuario (Haight & Bandopadhyay, 2015).

El análisis de la velocidad ayuda a determinar los tiempos de las tareas en el ciclo de desarrollo de software con DevOps.

La métrica de velocidad se encuentra ligada a tiempos Δt de los procesos que se desean tomar como referencia para analizar la efectividad en el ciclo de desarrollo de software.

El Tiempo Medio de Reparación (MTTRS) define la velocidad a la que se puede reparar el servicio luego de un fallo producido en cualquiera de los componentes antes descritos.

La resolución efectiva de los fallos permite disponibilidad de los servicios que se brinda a los usuarios.

El Tiempo de Ciclo representa la porción de tiempo que demora desde la solicitud de cambio o requerimiento en ser atendida hasta su implementación o paso a producción.

La Frecuencia de Implementación representa la cantidad de veces que se despliega a producción una aplicación al usuario.

¹⁷ Según la RAE

Entre las Métricas en el Análisis del Servicio que analizan la calidad del servicio está el Apoyo a la Producción y el índice de éxito de la implementación.

La Métrica de Apoyo a la Producción es la relación de las horas empleadas en solucionar problemas frente a las horas empleadas en el desarrollo de nuevos productos (Haight & Bandopadhyay, 2015).

La Métrica índice de éxito de la implementación considera el porcentaje de despliegues que han tenido una interrupción o fallo al ser utilizadas por el usuario.

El porcentaje de fallo en la etapa de posproducción da la pauta para entender si las fases de desarrollo se están cumpliendo en forma efectiva, de tal manera que el valor de errores debería ser mínimo.

El tiempo de restauración de los fallos si es mínimo no influye en la tasa de éxito de la implementación debido a que los errores deben resolverse en etapas definidas del ciclo de desarrollo DevOps.

Las Métricas en la Administración de Sistemas también conocido como Operaciones de TI tratan de determinar la eficiencia de los servicios que se entregan al usuario interno y cliente externo.

La Métrica de Usuarios FTE representa la cantidad de usuarios por Empleados a tiempo completo que se requieren para manejar todos sus requerimientos. Mientras más usuarios existan más empleados se necesitan para atender las necesidades de desarrollo de software.

La Métrica de Servidores FTE considera el número de servidores que se requiere para atender la demanda de desarrollo de aplicaciones por parte de todos los desarrolladores de aplicaciones. En un entorno de DevOps, este valor debe crecer a medida que se aplica al entorno una mayor automatización (Haight & Bandopadhyay, 2015).

La Métrica Costo por Transacción representa el costo por transacción es una métrica eficiente de los centros de datos que se centra en el costo de la energía por transacción y, por lo general, se mide en términos de kilovatios por transacción (Haight & Bandopadhyay, 2015).

La Métrica Costo del Cambio o Lanzamiento refleja los costos de cambios para satisfacer al usuario sea en etapa de despliegue o posproducción.

Se toma como referencia las siguientes métricas para esta tesis:

- Habilitación de nuevos servicios empresariales
- Net Promoter Score
- Satisfacción de los usuarios
- Colaboración, como principio de Integración Continua
- Tiempo de ciclo frecuencia de implementación

Estos parámetros al estar en relación con todos los componentes DevOps y con los componentes de cada una de las iteraciones en donde intervienen las etapas de desarrollo y despliegue de software, van a establecer lineamientos que ayudan a verificar el avance de los casos de estudios planteados.

6. ALINEANDO PLANIFICACIÓN ESTRATÉGICA, REQUERIMIENTOS Y SERVICIOS

En este capítulo se integran herramientas y metodologías para demostrar cómo transformar los requerimientos en servicios a disposición en tiempo y forma, siguiendo la planificación que se obtienen desde las estrategias.

El objetivo es optimizar la relación Organización-Usuarios en el marco de las buenas prácticas, del alineamiento estratégico, del uso de las metodologías ágiles, estableciendo una estructura organizacional del Data Center más dinámica en base a componentes DevOps.

Se integran los conceptos definidos y analizados en los capítulos anteriores para cumplir con el objetivo de esta tesis: proponer un modelo para la administración eficiente de recursos de un data center como resultado del alineamiento estratégico entre negocio y TI, utilizando metodologías de desarrollo ágiles, integrando desarrollo y despliegue y soportado por buenas prácticas reconocidas internacionalmente.

En ese contexto el enfoque planteado se presenta en base a la siguiente organización de elementos: una relación de elementos plasmados en una figura con su respectiva explicación, y una hoja de ruta que sirve de guía para que el modelo planteado sea fácilmente implantado para alcanzar una administración eficiente de recursos de un Data Center.

Se desarrollan dos casos de estudio, donde se aplica el modelo SAM-RS y las herramientas DevOps. Los resultados son evaluados en base a diferentes métricas DevOps.

6.1. Modelo orientado a la administración eficiente de recursos de un Data Center

La Figura 6.1 representa el enfoque gráfico del modelo que integra desarrolladores y operadores a partir de la concepción de DevOps, alineando componentes para la oferta de servicios corporativos, en un contexto de buenas prácticas en los servicios de TI. El gráfico se lo ha realizado en varios colores únicamente para visualizar cada componente del modelo planteado.

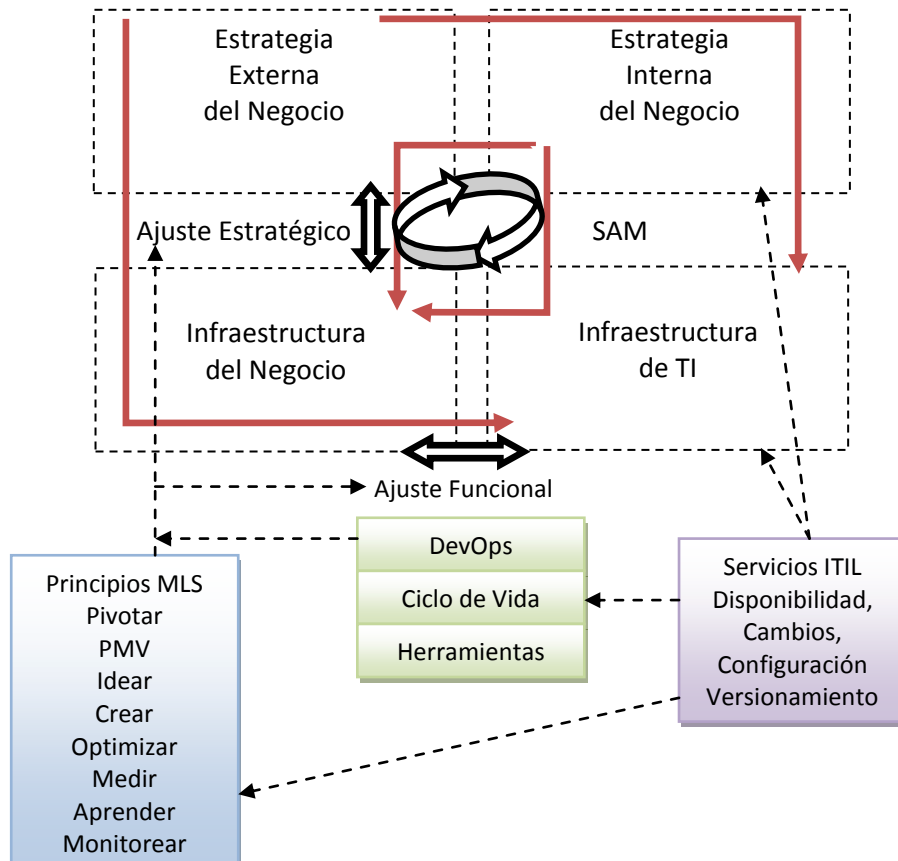


Figura 6.1 Estructura de componentes para un manejo eficiente de recursos de un data center

6.2. Relaciones gráficas del enfoque de modelado

La explicación de las relaciones mostradas en el gráfico anterior se lo explica en las siguientes secciones.

6.2.1. DevOps – MLS

El manejo adecuado de los principios en los que se basa DevOps permite una estructura colaborativa en todo el Ciclo de Vida de DevOps (Ver sección 4.3.2).

En la Figura 6.1 se visualiza a DevOps con el ciclo de vida y las herramientas de soporte. Los Principios de DevOps, Integración Continua, Entrega Continua, Despliegue Continuo, forman parte del ciclo de vida.

Las Herramientas DevOps ayudan a adoptar principios de DevOps, permitiendo un enfoque dinámico al ciclo de desarrollo y despliegue de aplicaciones.

Los Principios de MLS mantienen una relación directa con los componentes de DevOps (Ver Figura 6.1).

Desde la etapa de Planificación de la Iteración hasta la etapa de producción, el soporte de los Principios de MLS es estratégico. Establecen una dinámica entre el equipo DevOps y el usuario.

La relación entre principios de MLS y DevOps, influye en la calidad del desarrollo de software.

Cada principio de Lean Startup da soporte a cada etapa del Ciclo de Vida DevOps.

El principio de Producto Mínimo Viable apoya al Ciclo de Vida DevOps en la búsqueda de iteraciones cortas que permitan descifrar la incertidumbre que ocasiona la falta de definición de requerimientos.

Al construir cada iteración en corto tiempo se vislumbra más fácilmente los problemas que presenta el usuario al definir sus tareas a automatizar.

6.2.2. Relación DevOps - ITIL

Los servicios de ITIL, configuración, monitoreo y versionamiento tienen influencia directa con todas las etapas del ciclo de vida DevOps (Ver Figura 6.1).

La Gestión de Configuración se presenta como apoyo a las etapas del Ciclo de Vida DevOps, basados en construcción de aplicaciones en base a iteraciones cortas y PMV.

Desde la etapa de Planificación de la Iteración hasta el Despliegue de Aplicaciones, la Gestión de Configuración de ITIL se encarga del soporte de planificación que los operadores realizan para estructurar los distintos tipos de virtualización de acuerdo a la etapa en la que se encuentre el equipo de desarrollo.

La gestión de configuración garantiza el desarrollo de aplicaciones virtualizadas que permita despliegues automáticos del software estándar a ser utilizado por el equipo de desarrollo.

La Gestión de Configuración de ITIL permite administrar los recursos de Virtualización desde etapas tempranas del Ciclo de Vida DevOps.

La gestión de configuración garantiza que en la etapa de Gestión de Pruebas los operadores configuren máquinas virtuales con espacios y características similares a las de producción para que el equipo DevOps junto a los usuarios realice las respectivas pruebas de las aplicaciones.

La Gestión de Configuración apoya la definición y gestión de versionamiento de las aplicaciones. La gestión de versiones utiliza infraestructura de software y hardware para controlar los cambios que se van realizando en las aplicaciones.

La Gestión de Configuración realiza los soportes adecuados de pre-configuraciones y configuraciones en todas las etapas del desarrollo y despliegue de software con el Ciclo de Vida DevOps.

La Gestión de Configuración permite determinar las herramientas apropiadas para el manejo de integración continua, indispensables en un proceso manejo de DevOps.

La Gestión de Monitoreo que se establece como prioritario en el Data Center es soportado por la Gestión de Configuración de buenas prácticas de ITIL.

La Configuración de la Gestión de Monitoreo permite obtener logs con información precisa sobre la cual se realice análisis de la información reflejada con el fin de realizar mejoras y correctivos en los recursos de ti.

La configuración de la Gestión de Monitoreo que obtenga información basura o logs extensos que no aporten valor agregado, es decir que no sea posible lograr un análisis

que refleje los puntos de fallos que se puedan tener o anticipar posibles errores que puedan presentarse en las etapas de producción, hacen que se pierda un aliado fuerte en los componentes claves del Data Center.

La Gestión de Versionamiento brinda su aporte directo a los principios de DevOps de Entrega Continua y Despliegue Continuo.

En cada Entrega Continua y Despliegue Continuo se aportan a los usuarios por parte del Equipo DevOps versiones Candidatas. El control del despliegue de las versiones candidatas van desplegando es el propósito de la Gestión de Versionamiento.

El soporte de la Gestión de Versionamiento requiere de Herramientas DevOps que ayuden al control adecuado que se requiere al desplegar una versión Candidata.

Las herramientas DevOps brindan el soporte adecuado para agilizar el buen desempeño en las fases del ciclo de vida de desarrollo y despliegue con DevOps.

6.2.3. Relación MLS - SAM-RS

Los Principios de MLS ayudan a establecer un soporte adecuado al alineamiento entre Gobierno Corporativo y TI (Ver Figura 6.1).

Los Dominios del Modelo de Alineamiento Estratégico SAM-RS, permiten estructurar la estrategia de la organización desde varias aristas de apoyo al establecer un modelo organizacional en base a requerimientos y servicios que evidencia la relación entre los recursos del Negocio y TI.

Los Dominios de SAM-RS establecen relaciones y coordinaciones directas con la etapa de Planificación de Requerimientos a ser Automatizados.

La estrategia del uso de SAM-RS y MLS para establecer la recopilación de requerimientos en base a perspectivas que ofrece cada Dominio de SAM-RS, ayuda a la definición eficaz de necesidades a ser automatizadas en plazos dentro de cronogramas establecidos en la Planificación de Tareas.

El Principio de las Ideas de MLS consigue que los servicios que oferta la Organización al usuario, sean basados en ideas que le ayuden a ser competitivo.

El aporte de Ideas innovadores en ámbitos empresariales ayuda a establecer directrices de lineamientos técnicos, administrativos, económicos, relacionados directa o indirectamente con los servicios que se provee al usuario.

Los Principios Crear, Aprender, Optimizar, dan su aporte en la gestión empresarial.

Crear servicios para el usuario, implica organizar y optimizar los procesos administrativos, técnicos, y alineados que permitan dar valor agregado a las Competencias Distintivas de la Organización.

La necesidad de Aprender de los buenos servicios que impacten o de los servicios que no tienen mayor aporte al usuario, logra determinar los correctivos necesarios para cambiar aspectos escalares mínimos o de fondo para adecuar con costos razonables los errores que se presentan al brindar prestaciones no adecuadas.

El aporte de MLS provee un soporte adecuado a los Requerimientos a ser Automatizados con el fin de satisfacer las necesidades corporativas con valor agregado.

El Pivotaje que se realiza en los Dominios de SAM-RS, compromete las ayudas de las áreas involucradas en el análisis de requerimientos. Cada Dominio como dueño del proceso controla el Pivotaje para obtener información que no fue obtenida bajo el Control de obtención de requerimientos de otros Dominios.

6.2.4. Relación ITIL – Infraestructura interna de TI – Estrategia externa de TI

Los servicios que entrega ITIL permiten buenas prácticas y son transversales a la relación de infraestructura interna y estrategia externa de TI (Ver Figura 6.1).

La disponibilidad de recursos de TI que entrega TI al negocio debe estar soportada por infraestructura administrativa con procesos optimizados e infraestructura de TI alineados con los servicios que se provee a los usuarios.

La Gestión de Configuración tiene congruencia directa con el Dominio Infraestructura de TI, en donde se estipula todas las estrategias que han sido establecidas para dar soporte a la infraestructura administrativa logrando un Ciclo de Vida DevOps coherente en pre-configuraciones y configuraciones de acuerdo a la etapa en la que se encuentre el equipo desarrollador.

La gestión de cambios da soporte en las etapas de Planificación de Requerimientos a ser Automatizados y Gestión de Producción.

La Gestión de Cambios tiene su soporte con los Dominios de SAM-RS y los Principios de Lean Startup mientras dure la fase de los Cambios a realizar.

La Gestión de Cambios se presenta en base a las estrategias de apoyo que tiene el equipo DevOps desde la organización, apoyados con procesos optimizados e ideas que permitan requerimientos claramente definidos.

En este capítulo se presentan dos casos de estudio con el objetivo de aplicar los conceptos, metodologías y herramientas planteadas.

La aplicación de todos los elementos metodológicos que ayudan a optimizar el despliegue de aplicaciones como resultado del alineamiento estratégico de los servicios del Gobierno Corporativo y TI se evidencia en los siguientes casos de estudio.

En el primero se puede visualizar el axioma que no es posible implementar DevOps si no existe el adecuado alineamiento corporativo con TI.

En el segundo caso de estudio se puede apreciar el detalle con el que se deben granular los componentes del Data Center para lograr el control adecuado que permita disponibilidad de los servicios que están relacionados con las aplicaciones que utiliza el usuario.

6.3. Hoja de ruta del modelo orientado a la administración eficiente de recursos de un Data Center

El modelo orientado a la administración eficiente de recursos de un Data Center requiere la siguiente hoja de ruta:

- Determinar si una empresa tiene PE y GTI
- Determinar el plan estratégico corporativo (PE)
 - Planificación externa del negocio
 - Planificación interna del negocio
- Determinar el plan estratégico del gobierno de TI (GTI)
 - Planificación externa del GTI
 - Planificación interna del GTI
- Determinar ámbito de trabajo y necesidades estratégicas de la organización
- Definir personal administrativo representante de la alta gerencia para llevar adelante la implantación del PE
- Definir personal de TI representante de TI para llevar adelante el proceso de implantación de GTI.

- Determinar principios de MLS que sirvan de apoyo a las perspectivas
 - Pivotar
 - PMV
 - Idear
 - Innovar
 - Crear
 - Optimizar
 - Medir
 - Aprender
 - Monitorear

- Socializar y Capacitar el PE y GTI a personal administrativo y TI
- Capacitar a personal administrativo y TI en modelo de alineamiento estratégico y buenas prácticas
- Describir las estrategias externas e internas de la organización
- Determinar requerimientos externos e internos corporativos y de TI
- Determinar servicios externos e internos corporativos y de TI
- Aplicar proceso de alineamiento 1 con perspectivas (1) y (2)
- Aplicar proceso de alineamiento 1 con perspectivas (3) y (4)
- Aplicar proceso de alineamiento 2 con perspectivas (1) y (2)
- Aplicar proceso de alineamiento 2 con perspectivas (3) y (4)
- Análisis de Requerimientos alineados entre Dominios de SAM
- Planificar estrategias para enfrentar automatización de requerimientos
 - Planificar roles y ciclo de vida DevOps
 - Definir equipo de trabajo
 - Líder de equipo
 - Desarrollo
 - Operadores
 - Usuario
 - DevOps
 - Planificar tareas a automatizar
 - Organizar distribución de tareas
 - Planificar desarrollo de la Iteración
 - Roles y responsabilidades
 - Herramientas DevOps a utilizar
 - Desarrollo: Integración continua
 - Despliegue: Configuración
 - Control de Versiones
 - Control de Monitoreo
 - Métricas a utilizar (tiempo de ciclo)
 - Planificar pruebas de la iteración
 - Planificar ciclo de vida del despliegue de la iteración
 - Planificar control de software en producción
 - Activar Protocolo de Monitoreo en la barrera de la tolerancia a Fallos

- Determinar el objetivo del monitoreo
- Definir propiedades de apoyo de buenas prácticas
- Determinar la herramienta de monitoreo de acuerdo a sus propiedades
- Determinar el rango de tiempo del monitoreo en la barrera de la tolerancia de fallos
- Definir los componentes de software macro relacionados con la aplicación a monitorear
- Detallar los componentes hasta un nivel atómico (que ya no tiene subcomponentes)
- Aplicar el monitoreo sobre el componente definido
- Realizar pruebas de funcionamiento

Con esta hoja de ruta se canalizan los dos casos de estudio.

6.4. Caso de Estudio 1: Alineamiento Corporativo con TI

6.4.1. Determinar si una empresa tiene PE y GTI

En razón que SAM-RS se basa en la premisa, para aplicar su fundamento teórico es necesario que una organización tenga su planificación estratégica corporativa y TI definidas, se parte de un análisis realizado para focalizar las empresas que cumplan con este concepto.

De acuerdo a un relevamiento realizado con el apoyo de alumnos de la Facultad de Ingeniería, Ciencias Físicas y Matemática de la Universidad Central del Ecuador, Curso de Gestión de Proyectos, en el marco de análisis de planificación estratégica de varias empresas.

Se eligieron empresas públicas y privadas de la ciudad de Quito, con un mínimo de 500 empleados, para determinar algunos parámetros que permitan definir si las empresas tienen plan estratégico corporativo y TI definidos. Se escogieron 30 sobre las que se realizó una encuesta con los siguientes parámetros:

- a. Empresas que tengan un Plan Estratégico Corporativo (PE)
- b. Empresas que tengan una Planificación Informática TI (PGTI)
- c. Empresas con Altos Ejecutivos que conozcan de algún modelo o marco de referencia de alineamiento entre PE y PGTI
- d. Empresas con Ejecutivos de TI que conozcan de alineamiento entre PE y PGTI
- e. Empresas con Personal Operativo de la Organización que conozca de la existencia del PE
- f. Empresas con Personal de TI que conozcan de la existencia del PGTI
- g. Empresas con Personal que conozcan del PE y PGTI de su empresa

Los resultados obtenidos se presenta en la Tabla 6-1.

Tabla 6-1 Resultados evaluación 30 empresas

	A	B	C	D	E	F	G
Número de Empresas	20	28	10	8	4	4	2

El análisis de la tabla anterior es como sigue:

20 Empresas tienen Plan Estratégico Corporativo (PE)

28 Empresas tienen una Planificación Informática TI (PGTI)

10 Empresas tienen Altos Ejecutivos que conozcan de algún modelo o marco de referencia de alineamiento entre PE y PGTI

8 Empresas tienen Ejecutivos de TI que conocen de alineamiento entre PE y PGTI

4 Empresas tienen Personal Operativo de la Organización que conoce de la existencia del PE

4 Empresas tienen Personal de TI que conoce de la existencia del PGTI

2 Empresas tienen Personal que conoce del PE y PGTI de su empresa

Este estudio evidencia el estado de situación con respecto a la preponderancia que las organizaciones dan a su plan estratégico corporativo y de TI.

A los efectos de aplicar lo investigado en esta tesis, se optó por definir una empresa ficticia como caso de estudio.

Llamaremos a esta empresa “Empresa de Servicio de Energía Eléctrica” (ESEE).

Ante la intención de alinear la ejecución efectiva del Plan Estratégico Corporativo con el apoyo del Plan del Gobierno de TI se desarrolló el Proyecto de Modernización de la Empresa de Energía Eléctrica (PMESEE).

El PMESEE está determinado por:

- Plan estratégico corporativo (PE)
 - Planificación externa del negocio
 - Planificación interna del negocio
- El plan estratégico del gobierno de TI (GTI)
 - Planificación externa del GTI
 - Planificación interna del GTI

6.4.2. Determinar ámbito de trabajo y necesidades estratégicas de la organización

La empresa ESEE tiene como Misión proveer de energía eléctrica en forma continua a sus usuarios de acuerdo a los estándares regionales.

Tiene como Visión ser empresa Líder en la provisión continua del servicio de energía eléctrica.

La ESEE provee el servicio de energía eléctrica a una población urbana y rural, con más de un millón de abonados.

La ESEE al tener como visión la provisión continua de energía eléctrica a sus clientes, debe mantener óptimos todos los servicios relacionados, ya que la interrupción por alguna razón es perjudicial para la ESEE y para el Cliente.

6.4.2.1. Usuario como foco de Estudio

La ESEE posee dos tipos de clientes y usuarios de servicios:

- El Cliente que es el que paga por el servicio de energía eléctrica consumido
- El usuario interno de los servicios internos de la ESEE, entre ellos el tecnológico.

El usuario interno al que se refiere el caso de estudio 1, es el usuario consumidor de los servicios computacionales. Sobre sus actividades se van a ejercer tareas de modernización operativas que aportan los siguientes beneficios:

- Innovar los servicios con los que opera el usuario interno a fin de aumentar la disponibilidad del servicio de energía eléctrica al cliente externo.
- Alcanzar mejores ingresos económicos para la organización, se consigue al tener mínimas interrupciones de energía eléctrica.

6.4.2.2. Organización de la Empresa ESEE

En esta sección se presenta los siguientes aspectos:

- Personal administrativo representante de la alta gerencia para llevar adelante la implantación del PE
- Personal de TI representante de TI para llevar adelante el proceso de implantación de GTI.

La empresa ESEE se estructura en tres direcciones: Dirección Comercial, Dirección de Operaciones y Dirección de TICs.

Cada dirección tiene departamentos con jefaturas y personal administrativo y técnicos de acuerdo a su función dentro de la organización.

De acuerdo al PE de la ESEE, se consideró que la Dirección Comercial era la adecuada para asignarle la dirección de proyecto.

La Dirección Comercial consta de 3 departamentos, y cuenta con 300 empleados.

Dos de los departamentos tienen algún tipo de relación con atención al público.

En el Departamento de Análisis Administrativo, trabajan analistas administrativos que no atienden al público.

La Dirección de Tecnología da soporte de TI a toda la organización.

Está estructurada en tres departamentos, a saber:

- Dpto. de Infraestructura
- Dpto. de Desarrollo y Mantenimiento de Software
- Dpto. Administrativo y Soporte

Cuenta con 40 empleados que se distribuyen de la siguiente manera:

- 14 en el Dpto. de Infraestructura
- 17 en el Dpto. de desarrollo y mantenimiento
- 5 en Service Desk, parte del Dpto. Administrativo y soporte
- 4 en el área administrativa del Dpto. Administrativo y soporte

6.4.2.3. Identificación del Problema y Solicitud del Proyecto

Para modernizar sus servicios la ESEE ha estructurado el “Proyecto de Modernización de la Empresa de Servicio de Energía eléctrica PMESEE”, el cual tiene como objetivo apoyar la ejecución efectiva del Plan Estratégico Corporativo con el apoyo del Plan del Gobierno de TI.

Se tomó como referencia para este caso de estudio el siguiente eje estratégico del PMESEE “Modernización del servicio de reconexión de energía eléctrica”.

Sobre este eje se van a identificar todos los procesos relacionados con la provisión, corte y reconexión del servicio de energía eléctrica. TIC va a dar soporte con sus recursos en el objetivo de disminuir las interrupciones del servicio de energía eléctrica optimizando la prestación de energía eléctrica al cliente externo.

En el PMESEE se solicita modernizar el servicio de reconexión de energía eléctrica.

La causa que promueve esta petición es la siguiente:

- El corte de energía eléctrica se produce por la falta de pago de la factura por parte del cliente.

Se ha detectado que al realizar el corte de energía eléctrica por parte del operador en el medidor del cliente, se pierde considerables ingresos económicos por falta de consumo del servicio de energía eléctrica. Se considera que la penalización de corte también afecta a la empresa y causa molestia al usuario.

- La empresa deja de percibir ingreso durante el tiempo que permanece el cliente sin energía eléctrica.
- El cliente externo se queda sin energía eléctrica, servicio básico en su vida diaria, con los problemas que esta situación conlleva, que en algunos casos pueden ser críticos.

Puede ser que el cliente externo vea el corte de energía por falta de pago como una molestia grande, a pesar que fue notificado de la fecha de vencimiento de su pago, mediante correo electrónico, mensajes a celular.

De la misma manera antes del corte ya fue notificado que se va a proceder con la suspensión del servicio por no haber cancelado su factura.

Los dos puntos son razones suficientes para promover la modernización del servicio de reconexión de energía.

Con respecto al primer punto en la Tabla 6-2, se identifican las causas fundamentales por la cuales se debe resolver el problema.

Tabla 6-2 Número de cortes de energía promedio mensual por falta de pago

Número de Cortes Promedio Mensual	Tiempo promedio de reconexión	Valor kw-hora (dólar)	Kw promedio consumidos por hora en sector residencial	Valor por 3 horas no consumidas (dólar)
29.000	3 horas	0,077	1,45	9713,55

El número de cortes promedio mensual, se refiere a que existe un promedio de 29.000 clientes externos que se les ha cortado el servicio de energía eléctrica por falta de pago de su factura.

El tiempo promedio de reconexión se explica de la siguiente manera:

- Una vez realizado el corte de energía, el cliente externo debe cancelar su factura para que pueda volver a ser reconectado el servicio.
- Mientras no pague el cliente externo no tendrá energía eléctrica.
- Este tiempo no es tomado en cuenta en este estudio de caso.
- Existen dos turnos del personal de cortes y reconexiones, uno desde las 07:00 hasta las 14:00 y el segundo desde las 13:00 hasta las 20:00.
- El listado de cortes y reconexiones para el turno de la mañana es entregado a cada operador a las 08:00.
- El listado de cortes y reconexiones se entrega a partir de las 15:00.
- El momento en que el cliente cancela su factura, se actualiza en la base de datos su pago.

- El listado de los clientes que pagaron su factura pero ya fueron cortados la energía es entregado en los turnos de la mañana y en la tarde. Es decir si se pagó en la mañana, su servicio será reconectado en la tarde.
- También existe la posibilidad de que el personal de cortes que se encuentran en las oficinas, monitoreen los pagos realizados por los clientes externos y llamen por radio a los supervisores de corte y estos a su vez a los operadores para que realicen la reconexión respectiva a ese cliente.
- El personal de corte de la mañana sale a realizar su trabajo con el listado entregado a las 08:00, que corresponde a los clientes externos que no pagaron su factura hasta el día anterior al corte.
- Puede darse el caso que el operador se encuentra en el sitio de corte y el cliente externo ya pagó su factura a la misma hora. El operador realiza el corte ya que no tiene evidencia que se haya realizado el pago.
- El tiempo promedio se calcula desde que el cliente externo pagó su factura hasta la reconexión respectiva por parte del operador.

En la Tabla 6-2 se puede apreciar que los valores son significativos debido al alto número de cortes. No se considera en esta tesis el tratamiento de las estrategias necesarias para reducir este número de cortes.

Se puede apreciar que es preocupación de los ejecutivos de la organización la disminución de valores económicos que se generan debido a los espacios de tiempos sin consumo de energía eléctrica desde el corte hasta su reconexión.

La provisión de energía eléctrica es fundamental para las labores diarias de los clientes. Para muchos clientes externos puede ser clave pocos minutos sin el servicio de energía eléctrica a causa de un corte por falta de pago, razón fundamental, no fácilmente cuantificable, para disminuir los tiempos de corte y reconexión del servicio de energía eléctrica.

6.4.3. Definir personal administrativo representante de la alta gerencia para llevar adelante la implantación del PE

Personal de la Dirección Comercial, encargado de llevar adelante el inicio del proyecto.

- 3 personas del Dpto. de Análisis Administrativa de la dirección comercial.
- 3 operadores del área de cortes y reconexiones. (Personal que se encarga del corte de energía en el medidor el cliente externo por falta de pago).

6.4.4. Definir personal de TI representante de TI para llevar adelante el proceso de implantación de GTI

La Dirección de Tecnología puso a disposición del PMESEE, es decir a las 40 personas de los 3 departamentos, sin embargo no todos fueron tomados en cuenta para el proyecto.

Personal de la Dirección de TI encargado de soporte en el inicio del proyecto.

- Dpto. Administrativo y soporte

- Número de personas 6

Fechas del proyecto: Abril 2016 – Septiembre de 2016.

6.4.5. Determinar principios de MLS que sirvan de apoyo a las perspectivas

Los principios de MLS a tomar en cuenta serán basados en la estructura que se presente en las diferentes perspectivas. El principio de pivotaje servirá de guía en la búsqueda de requerimientos del usuario. Los principios de Idear, Innovar, Aprender, Optimizar generan espacios de conocimiento para definir procesos acorde a las necesidades de los usuarios y ayudan a alinear los objetivos corporativos con los recursos de TI.

6.4.6. Socializar y Capacitar el PE y GTI a personal administrativo y TI

Conforme a los principios de MLS que propicia la uniformidad e intercambio de conocimiento, se capacitó al personal involucrado (12 personas) en los siguientes elementos: PE y GTI.

6.4.7. Capacitar a personal administrativo y TI en modelo de alineamiento estratégico y buenas prácticas

Se ha capacitado al personal administrativo y TI en los siguientes elementos:

- Modelo SAM-RS,
- Metodología MLS
- Con el fin de enfocar el análisis de requerimientos de acuerdo a prácticas internacionales probadas se capacitó al personal de la Dirección de Tecnología en: ITIL y Métricas DevOps

Con respecto a las herramientas DevOps, el equipo DevOps ya tenía experiencia en su uso, por lo tanto no requirió capacitación.

6.4.8. Describir las estrategias externas e internas de la organización

Para realizar una recopilación adecuada de requerimientos y definir las verdaderas necesidades de automatización del usuario, el Caso de Estudio 1 toma como base las perspectivas descritas en las Figura 3.8 y Figura 3.9, que describen la relación de estrategias de los dominios de SAM.

Se empieza con los procesos 1 y 2 y las perspectivas de alineamiento (1), (2), (3) y (4).

Son utilizadas para definir las estrategias, servicios y requerimientos de cada uno de los dominios, con miras a modernizar las actividades del PMESEE que apoya la ejecución efectiva del Plan Estratégico Corporativo con el apoyo del Plan de Gobierno de TI.

Se plantean dos procesos.

El proceso 1 involucra las actividades inherentes a las perspectivas (1) y (2) del modelo SAM, según se analizó en la sección 3.1.2.

El proceso 2 involucra las actividades inherentes a las perspectivas (3) y (4) del modelo SAM, según se analizó en la sección 3.1.3.

La planificación para determinar los alcances y competencias de cada dominio, y definir el ajuste estratégico, sigue el enfoque desde la estrategia hacia los recursos.

Las relaciones entre dominios tienen secuencia jerárquica, es decir los dominios se subordinan a la estrategia cuyo dominio se encuentra llevando el control de la planificación de tareas de modernización empresarial.

La Planificación externa se la realiza con el fin de optimizar procesos que ayuden a la entrega efectiva de servicios a los clientes externos.

La Planificación Interna se la lleva adelante con el fin de utilizar recursos adecuados y procesos optimizados en su afán de modernizar la organización en forma coordinada con los otros dominios.

La Tabla 6-3 describe las estrategias externas e internas según el PMESEE, de acuerdo al SAM.

Tabla 6-3 Entradas y Salidas de estrategias externas e internas del negocio y TI

Cód EN01	Estrategia del Negocio Modernización del servicio de reconexión de energía eléctrica		
PEX	Estrategia Externa del Negocio	GTIEX	Estrategia Externa de TI
PEX01	- Proteger los intereses de la empresa y sus accionistas mediante la modernización de los servicios entregados al usuario.	GTIEX01 GTIEX02	- Analizar tácticas con procesos tecnológicos que ayuden a modernizar la organización. - Analizar la colaboración que el conjunto de sistemas y capacidades de TI brindan como soporte en el marco del negocio.
PIN	Estrategia Interna del Negocio	GTIN	Estrategia Interna de TI
PIN01	- Determinar el flujo de procesos adecuado en el marco de la provisión de servicio de reconexión de energía eléctrica.	GTIN01	- Coordinar acciones para asignar recursos de TI que den soporte a la Estrategias Externas de GTI y Estrategias Internas del Negocio.
PIN02	- Coordinar acciones con Estrategias Externas e Internas de TI, a fin de que los recursos administrativos y TI sirvan de soporte a los servicios de reconexión de energía eléctrica.		

6.4.9. Determinar Requerimientos externos e internos de la organización

La definición y aplicación de requerimientos se presenta como resultado del análisis de SAM-RS expuesto en la sección 3.2 y en la Figura 3.10.

Los requerimientos definen las necesidades que tiene cada dominio para cambiar su entorno en base a estrategias definidas para modernizar los ámbitos relacionados con la entrega de servicios a los usuarios.

Las necesidades de los requerimientos varían de acuerdo a las relaciones que se vayan dando entre dominios y en concordancia con los servicios que estratégicamente se vayan diseñando para ofrecer mejores prestaciones a los usuarios.

El prefijo “R” refiere a requerimientos. Por ejemplo, RPEX01, es la denominación del requerimiento en el contexto de la planificación PEX01 (Ver Tabla 6-4).

Tabla 6-4 Requerimientos Externos e Internos del Negocio y TI

RPEX	Requerimientos de apoyo a servicios Externos del Negocio	RGTIEX	Requerimientos de apoyo a servicios Externos de TI
RPEX01 RPEX02	<ul style="list-style-type: none"> • Determinar pautas para dar fluidez a la entrega de Servicio de energía eléctrica al usuario. • Optimizar procesos para disminuir tiempos de reconexión de servicio de energía eléctrica. 	RGTIEX01 RGTIEX02	<ul style="list-style-type: none"> • Determinar los recursos de TI relacionados con la entrega del Servicio de energía eléctrica • Determinar los recursos de TI que se necesitan para modernizar los Servicios de Reconexión de Servicio de Energía eléctrica.
RPIN	Requerimientos de apoyo a Servicios Internos del Negocio	RGTIN	Requerimientos de apoyo a servicios Internos de TI
RPIN01 RPIN02	<ul style="list-style-type: none"> • Coordinar la Definición y Optimización de procesos relacionados con los servicios de reconexión de energía eléctrica. • Canalizar acciones de alineamiento entre infraestructura administrativa e Infraestructura de TI con el fin de entregar procesos optimizados para la modernización del proceso de reconexión de energía eléctrica 	RGTIN01 RGTIN02	<ul style="list-style-type: none"> • Coordinar acciones para desarrollar y desplegar aplicaciones eficientes que permitan la modernización del servicio de reconexión de energía eléctrica. • Coordinar acciones para desarrollar aplicaciones que ayuden a modernizar los procesos optimizados por infraestructura administrativa.

6.4.10. Determinar Servicios externos e internos corporativos y TI

La definición y aplicación de servicios se presenta como resultado del análisis de SAM-RS expuesto en la sección 3.2 y en la Figura 3.10.

Con el fin de disminuir la brecha habitual que existe dentro de las organizaciones entre el gobierno corporativo y el gobierno de TI, surge la necesidad de identificar claramente los servicios que provee cada dominio, y determinar su fortaleza y participación en el servicio que se brinda al usuario.

El prefijo “SE” refiere a servicios. Por ejemplo, SEPEX01, es la denominación del servicio externo en el contexto de la planificación PEX01 (Ver Tabla 6-5).

Tabla 6-5 Servicios Externos e Internos del Negocio y TI

SEPEX	Servicios Externos del Negocio	SEGTIEX	Servicios Externos de TI
SEPEX01 SEPEX02	<ul style="list-style-type: none"> - Servicio de provisión de energía eléctrica al Cliente. - Servicio de reconexión de servicio de energía eléctrica 	SEGTIEX01	<ul style="list-style-type: none"> • Servicios de TI con Disponibilidad de acuerdo a niveles de servicio solicitados por el usuario.

		SEGTIEX02	<ul style="list-style-type: none"> Personal con las capacidades requeridas, de acuerdo a la definición de la tecnología adecuada conforme las pautas de la gobernabilidad del negocio.
		SEGTIEX03	<ul style="list-style-type: none"> Servicios de infraestructura como conjunto y combinación de aplicaciones, datos e información.
SEPIN	Servicios Internos del Negocio	SEGTIN	Servicios Internos de TI
SEPIN01	<ul style="list-style-type: none"> Flujo de procesos óptimo más adecuado para dar soporte a los objetivos estratégicos externos e internos. 	SEGTIN01	<ul style="list-style-type: none"> Servicios de Software
SEPIN02	<ul style="list-style-type: none"> Optimización de recursos mediante la coordinación de mejoras en los procesos y servicios de Infraestructura administrativa con costes mínimos. 	SEGTIN02	<ul style="list-style-type: none"> Servicios Computacionales
SEPIN03	<ul style="list-style-type: none"> Canalizar que los procesos que requiere automatizar TI, se entreguen optimizados y en forma oportuna. 	SEGTIN03	<ul style="list-style-type: none"> Servicios Mecánicos y Eléctricos
		SEGTIN04	<ul style="list-style-type: none"> Servicio de procesos propios de TI como mantenimiento, desarrollo, control, monitoreo.

Los procesos y perspectivas del modelo SAM-RS aplicadas a las estrategias, servicios y requerimientos empresariales de ESEE, se explican a continuación en cada proceso de alineamiento.

6.4.11. Problemas al iniciar la implementación de SAM-RS

En abril del 2016 se empezó el proyecto con la identificación de los requerimientos de la planificación externa e interna de los 4 dominios, de negocio y TI.

Este proceso tomó un mes y no reflejó mayor resultado, es decir no se encontraba evidencia de los requerimientos a ser analizados.

En las reuniones se culpaba al modelo SAM-RS que era muy ambiguo su funcionamiento. Luego de un breve análisis se detectó que no era el modelo SAM-RS el problema, era el personal que no estaba familiarizado con el plan estratégico corporativo y de TI.

A pesar que la encuesta reflejada en la Tabla 6-1, mantenía que si había personal conocía del plan estratégico corporativo y TI, ya en la práctica no era tan consistente su conocimiento para la implementación.

Se pasó a realizar un repaso muy detenido de los dos planes con las 12 personas involucradas en el proyecto. Este tiempo duró todo el mes de mayo de 2016. Superado este inconveniente se procedió a la implementación de SAM-RS.

6.5. Aplicar proceso de Alineamiento (1) Perspectiva (1) y (2)

Se parte del proceso de alineamiento (1) perspectiva (1).

El proceso de alineamiento (1) perspectiva (1) se muestra en la Figura 3.8, y sirve de base para el tratamiento de cambio de PEX – PIN- GTIN. Las estrategias del PMESEE con sus servicios y requerimientos, deben estar definidos.

PEX toma el control del análisis de la modernización corporativa.

6.5.1. Proceso de Alineamiento (1) Perspectiva (1) PEX-PIN

La alineación PEX-PIN (Ver Figura 6.2) permite definir la relación entre la Planificación, Requerimientos y Servicios del Plan Estratégico Corporativo externas e internas de la organización (Ver Tabla 6-6).

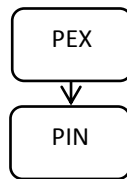


Figura 6.2 Bloque de Dominios Perspectiva (1) PEX-PIN

Tabla 6-6 Alineación PEX-PIN Original

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PEX	PEX01	RPEX	RPEX01 RPEX02	SEPEX	SEPEX01 SEPEX02
PIN	PIN01 PIN02	RPIN	RPIN01 RPIN02	SEPIN	SEPIN01 SEPIN02 SEPIN03

Para cumplir con el proceso inicial de modernización de PEX y PIN se definen requerimientos externos RPEX e internos RPIN que ayudan a definir las necesidades de cambios en la organización.

Los requerimientos externos e internos activan servicios externos SEPEX e internos SEPIN que brindan soporte adecuado a los cambios solicitados por los dominios para modernizar la organización en la entrega de prestaciones externas e internas a los usuarios.

Las estrategias internas de PIN mantienen la jerarquización de PEX, es decir PEX lleva el control de los cambios, y PIN apunta sus estrategias principales de apoyo hacia ese objetivo.

Las estrategias internas de PEX y PIN, mantienen la relación diagonal entre dominios.

En la relación diagonal, cada estrategia de PEX tiene apoyo de GTIN y PIN de GTIEX directa o indirectamente, de esta manera se alcanza la máxima relación entre ellos ayudando a establecer la conexión de soporte en forma directa y detallada de procesos.

Como resultado del análisis de las estrategias de cambios solicitados por PEX se desprende que es necesario crear una nueva estrategia interna de PIN para alcanzar mejor alineamiento entre la modernización de PEX y el soporte de PIN (Ver Figura 6.3).

- *PIN03: Coordinar mejoras en los servicios de Infraestructura administrativa que permita optimización de recursos con costes mínimos.*
- *Resultado: PIN se actualiza y se convierte en PIN´*

Figura 6.3 Establecimiento de nueva estrategia de apoyo interno PIN03

La nueva estrategia interna de apoyo PIN03 determina si se generan nuevos requerimientos y servicios que sirvan para cumplir el nuevo objetivo planteado.

Al existir cambios en la estrategia de PIN, la modernización inicial planteada ya no es la misma y se convierte en PIN´.

Cambia la estrategia interna, sin embargo Requerimientos y Servicios siguen siendo los mismos mientras no se analice si es necesario modificarlos de acuerdo a la necesidad de la estrategia PIN´.

Luego del análisis respectivo se desprende que PIN´ se diseña para reforzar las estrategias internas y no genera requerimientos y servicios adicionales.

Se aprecia que PEX mantiene su acostumbrado estatus de conformidad, al no estar acostumbrado a una recopilación de requerimientos que ayuden a identificar necesidades de automatización. Por ello su cambio radica en la identificación de estrategias.

La Tabla 6-7 presenta los cambios realizados donde se generan una nueva estrategia, modificando la estrategia interna original.

Tabla 6-7 Actualizada Estrategia nueva PIN03 con generación de PIN´

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PEX	PEX01	RPEX	RPEX01 RPEX02	SEPEX	SEPEX01 SEPEX02
PIN´	PIN01 PIN02 PIN03	RPIN	RPIN01 RPIN02	SEPIN	SEPIN01 SEPIN02 SEPIN03

El proceso de alineamiento (1) perspectiva PIN-GTIN se actualizó como se muestra en la Tabla 6-7, por lo tanto PIN ya no es PIN es PIN´.

6.5.2. Proceso de Alineamiento (1) Perspectiva (1) PIN´-GTIN

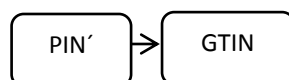


Figura 6.4 Bloque de Dominios Perspectiva (1) PIN´-GTIN

El nuevo PIN´ se analiza para verificar los tipos de apoyo que puede encontrar en la planificación que tiene el GTIN. GTIN aporta a la modernización de PIN´ de acuerdo a sus competencias y servicios, buscando relaciones directas de apoyo (Ver Figura 6.4) (Ver Tabla 6-8). PIN´ maneja los cambios y GTIN sirve de apoyo. Al realizar el respectivo análisis de estrategia-apoyo, se genera la siguiente nueva estrategia de soporte para PIN´. (Ver Figura 6.5).

- *GTIN02: Definir la tecnología adecuada de acuerdo a las pautas de gobernabilidad del negocio.*
- *Resultado: GTIN se actualiza y se convierte en GTIN'*

Figura 6.5 Establecimiento de nueva estrategia GTIN02 para Gobierno Interno de TI

Tabla 6-8 PIN' - GTIN original

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PIN'	PIN01 PIN02 PIN03	RPIN	RPIN01 RPIN02	SEPIN	SEPIN01 SEPIN02 SEPIN03
GTIN	GTIN01	RGTIN	RGTIN01 RGTIN02	SEGTIN	SEGTIN01 SEGTIN02 SEGTIN03 SEGTIN04

Al ser actualizado PIN en PIN', GTIN se transforma en GTIN' debido a que GTIN original debe incluir en su planificación el soporte respectivo de PIN', y GTIN debe actualizar sus procesos.

En la Tabla 6-9 se presenta GTIN' que es el resultado de la estrategia actualizada.

Tabla 6-9 Estrategia actualizada para GTIN'

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PIN'	PIN01 PIN02 PIN03	RPIN	RPIN01 RPIN02	SEPIN	SEPIN01 SEPIN02 SEPIN03
GTIN'	GTIN01 GTIN02	RGTIN	RGTIN01 RGTIN02	SEGTIN	SEGTIN01 SEGTIN02 SEGTIN03 SEGTIN04

No se realizaron nuevos requerimientos y servicios para enfrentar los cambios que GTIN' propone.

6.5.3. Proceso de Alineamiento (1) Perspectiva (2) PEX-GTIEX-GTIN'

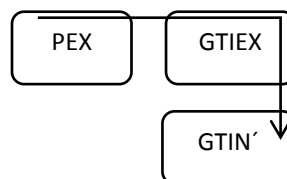


Figura 6.6 Bloque de Dominios con proceso 1 con perspectiva (2) con GTIN modificado

El Proceso de Alineamiento Proceso (1) Perspectiva PEX –GTIEX - GTIN se plantea con el PE y GTI externos e internos definidos y actualizados (Ver Figura 6.6).

6.5.4. Proceso de Alineamiento (1) Perspectiva (2) PEX-GTIEX

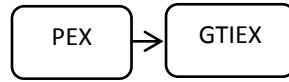


Figura 6.7 Bloque de Dominios Perspectiva (2) PEX-GTIEX

En la Figura 6.7 se puede visualizar que PEX y GTIEX no han sufrido variación en su planificación original.

De acuerdo al análisis de estrategias que realice PEX, GTIEX puede variar en GTIEX, SEGTIEX y RGTIEX de acuerdo al soporte que PEX solicite a GTIEX.

GTIEX luego de analizar el PEX, no deriva variaciones en su planificación, como tampoco genera requerimientos y servicios (Ver Tabla 6-10).

Tabla 6-10 PEX y GTIEX sin variación

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PEX	PEX01	RPEX	RPEX01 RPEX02	SEPEX	SEPEX01 SEPEX02 SEPEX03
GTIEX	GTIEX01 GTIEX02	RGTIEX	RGTIEX01 RGTIEX02	SEGTIEX	SEGTIEX01 SEGTIEX02 SEGTIEX03

6.5.5. Proceso de Alineamiento (1) Perspectiva (2) GTIEX - GTIN'

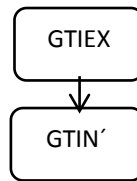


Figura 6.8 Bloque de Dominios Perspectiva (2) GTIEX-GTIN'

La Figura 6.8 y la Tabla 6-11, presentan las estrategias, requerimientos y servicios con los que se realiza el análisis para la perspectiva 2, que fueron tratadas en la sección 3.1.2.

Tabla 6-11 GTIEX y GTIN' con servicios y requerimientos antes de análisis de GTIEX

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
GTIEX	GTIEX01 GTIEX02	RGTIEX	RGTIEX01 RGTIEX02	SEGTIEX	SEGTIEX01 SEGTIEX02 SEGTIEX03
GTIN'	GTIN01	RGTIN	RGTIN01	SEGTIN	SEGTIN01

	GTIN02		RGTIN02		SEGTIN02 SEGTIN03 SEGTIN04
--	--------	--	---------	--	----------------------------------

GTIN' puede volver a variar de acuerdo a los requerimientos de GTIEX. Al analizar GTIEX se verifica que existen variaciones en su planificación estratégica interna. (Ver Figura 6.9).

- *GTIN03: Coordinar acciones que permitan disminuir costos de infraestructura de TI en base a alineación de requerimientos-infraestructura administrativa-infraestructura de TI.*
- *Resultado: GTIN' se actualiza y se convierte en GTIN''*

Figura 6.9 Establecimiento de nueva estrategia GTIN03 para Gobierno Interno de TI

Luego de ser actualizada la nueva estrategia GTIN03, la nueva estructura queda como se visualiza en la Tabla 6-12.

Tabla 6-12 GTIEX y GTIN' con su nueva estrategia GTIN03

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
GTIEX	GTIEX01 GTIEX02	RGTIEX	RGTIEX01 RGTIEX02	SEGTIEX	SEGTIEX01 SEGTIEX02 SEGTIEX03
GTIN''	GTIN01 GTIN02 GTIN03	RGTIN	RGTIN01 RGTIN02	SEGTIN	SEGTIN01 SEGTIN02 SEGTIN03 SEGTIN04

A continuación se analiza el Proceso de Alineamiento (2) Perspectiva (3)-(4) GTIEX-PEX-PIN y GTIEX - GTIN'' - PIN' a fin de analizar, si existen cambios en las estrategias de los dominios externos e internos, tanto del negocio como TI.

Es posible que en el Proceso de Alineamiento (2) se puedan presentar modificaciones en las planificaciones estratégicas, así como en los servicios y requerimientos de cada uno de los dominios.

El Pivotaje f1 que se plantea en la sección 2.3 de la Metodología Lean Startup permite analizar y replantear desde el punto de vista corporativo las estrategias, requerimientos y servicios a fin de determinar si existen cambios en la modernización corporativa a fin de presentar prestaciones con valor agregado al usuario.

El Pivotaje f1 se aplica al Proceso de Alineamiento (2) Perspectiva (3)-(4) GTIEX – PEX - PIN' y GTIEX-GTIN'' - PIN' para definir con más precisión la relación estrategias - requerimientos-servicios (Ver Figura 6.10).

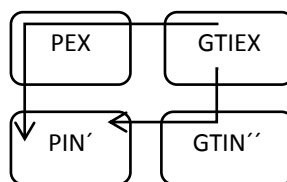


Figura 6.10 Proceso de Alineamiento (2) Perspectiva (3)-(4) GTIEX – PEX - PIN' y GTIEX - GTIN'' - PIN'

6.6. Aplicar proceso de Alineamiento (2) Perspectiva (3) y (4)

Se parte del proceso de alineamiento (2) perspectiva (3)

La estrategia de PMESEE con Proceso de Alineamiento Proceso (2) Perspectiva (3) GTIEX – PEX - PIN´ ya tienen actualizados sus respectivos planes estratégicos externos e internos (Ver Figura 6.11).

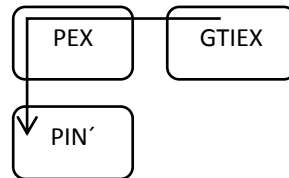


Figura 6.11 Bloque de Dominios Proceso de Alineamiento (2) Perspectiva (3) GTIEX-PEX-PIN´

GTIEX toma el control del proceso de cambio desde la perspectiva en la que el GTI de acuerdo a sus competencias debe manejar la Planificación Estratégica de ESEE (Ver Figura 6.12).

6.6.1. Proceso de Alineamiento (2) Perspectiva (3) GTIEX-PEX



Figura 6.12 Proceso de Alineamiento (2) Perspectiva (3) GTIEX-PEX

En la perspectiva GTIEX - PEX, TI mantiene el control de transformación corporativa. GTIEX y PEX tienen sus estrategias externas definidas.

En vista que TI mantiene el control del PMESEE, revisa las estrategias de PEX, y ahora PEX puede cambiar de acuerdo a lo estipulado por el análisis de GTIEX.

Al analizar el PEX por parte del GTIEX, se verifica que el PEX puede mejorar en la entrega de sus servicios al usuario con apoyo de los servicios de TI.

Desde la óptica de TI, la modernización de PEX se presenta en base a la innovación de las prestaciones tradicionales que ofrece la organización a sus usuarios.

Al realizar el análisis desde la perspectiva de GTIEX, se revisan los servicios que TI puede ofertar para mejorar las actividades del usuario.

En ese contexto se verifica que los servicios de TI para dispositivos móviles pueden ser utilizados por los usuarios internos de la organización en varios trabajos operativos, y pueden servir como apoyo directo en el cumplimiento de la modernización de servicios de energía eléctrica por el PMESEE.

Desde GTIEX se visualiza que los dispositivos móviles ayudan al personal operativo a dar mayor fluidez las labores y los requerimientos que se pueden presentar en un momento determinado relacionado con las actividades de entrega, corte y reconexión del servicio de energía eléctrica.

El oportuno apoyo que pueda tener el personal operativo en el campo, a través de medios computacionales servirá para atender con prontitud novedades existentes en el servicio de energía eléctrica.

Por lo tanto la estrategia de PEX en base a las nuevas estrategias definidas con ópticas de TI expresadas mediante el GTIEX, queda planteada en la Figura 6.13.

- *PEX02: Coordinar acciones estratégicas para mejorar servicios corporativos a través de tecnologías modernas.*
- *Resultado: PEX se actualiza y se convierte en PEX'*
- RPEX03: Determinar el alcance de entrega de nuevos servicios a través de nuevas tecnologías.
- SEPEX04: Servicios de Atención a usuarios mediante dispositivos móviles
- *GTIEX03: Coordinación entre las estrategias internas y externas del negocio, así como estrategias internas de TI para estructurar infraestructuras de apoyo administrativas y tecnológicas con servicios móviles*
- *Resultado: GTIEX se actualiza y se convierte en GTIEX'*
- RGTIEX03: Determinar Procesos Externos e Internos adaptados y optimizados a las nuevas tecnologías móviles que ofrece TI.
- SEGTIEX04: Servicios móviles de TI que ofrezcan disminución de tiempos y costos en las tareas administrativas de usuarios internos y externos.

Figura 6.13 Establecimiento de nuevas estrategias luego de analizar GTIEX' sobre PEX

Este conjunto de nuevas estrategias, requerimientos y servicios surge al interactuar los componentes que son parte del PEX, como los clientes externos con personal de TI encargados de llevar adelante el GTIEX.

La necesidad de mantener disponible el servicio de energía eléctrica al cliente externo, activa servicios mediante dispositivos móviles como estrategia inmediata para solventar este requerimiento.

La Tabla 6-13 presenta las nuevas estrategias, requerimientos y servicios luego de analizar GTIEX sobre PEX.

Tabla 6-13 Influencia de GTIEX sobre PEX resultando PEX'

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
GTIEX'	GTIEX01 GTIEX02 GTIEX03	RGTIEX'	RGTIEX01 RGTIEX02 RGTIEX03	SEGTIEX'	SEGTIEX01 SEGTIEX02 SEGTIEX03 SEGTIEX04
PEX'	PEX01 PEX02	RPEX'	RPEX01 RPEX02 RPEX03	SEPEX'	SEPEX01 SEPEX02 SEPEX03 SEPEX04

6.6.2. Proceso de Alineamiento (2) Perspectiva (3) PEX' - PIN'

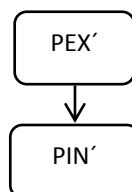


Figura 6.14 Proceso de Alineamiento (2) Perspectiva (3) PEX' - PIN'

En la perspectiva PEX' - PIN', PEX' cambió su estrategia de entrega de servicios externos, por lo tanto PIN' debe acoplarse a los nuevos requerimientos de PEX' (Ver Figura 6.14). Las nuevas estrategias de PEX' para entregar nuevos servicios con tecnologías modernas, obligan a PIN' a establecer nuevas estrategias interna y externas para diseñar nuevos flujos de procesos que ayuden a las estrategias externas e internas de TI a mejorar sus prestaciones.

Por lo tanto PIN' genera las siguientes estrategias, servicios y requerimientos (Ver Figura 6.15):

- PIN03: Coordinar definición de Procesos optimizados para Provisión de Servicios mediante Dispositivos Móviles
- Resultado: PIN' se actualiza y se convierte en PIN''
- RPIN03: Automatizar procesos optimizados para provisión de servicios mediante dispositivos móviles
- SEPIN04: Provisión de procesos optimizados para entrega de prestaciones mediante dispositivos móviles

Figura 6.15 Establecimiento de nuevas estrategias luego de analizar PEX' sobre PIN'

En la Tabla 6-14 se presentan la nueva estrategia, servicios y requerimientos luego de analizar PEX' sobre PIN'.

Tabla 6-14 Servicios y requerimientos nuevos luego análisis de PEX' sobre PIN'

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
PEX'	PEX01 PEX02	RPEX'	RPEX01 RPEX02 RPEX03	SEPEX'	SEPEX01 SEPEX02 SEPEX03 SEPEX04
PIN''	PIN01 PIN02 PIN03	RPIN'	RPIN01 RPIN02 RPIN03	SEPIN'	SEPIN01 SEPIN02 SEPIN03 SEPIN04

6.7. Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN'' - PIN''

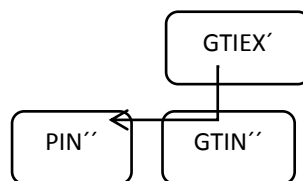


Figura 6.16 Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN'' - PIN''

En la estrategia de PMESEE con Proceso de Alineamiento Proceso (2) Perspectiva GTIEX' - GTIN'' - PIN'', ya se tienen actualizados los respectivos planes estratégicos externos e internos (Ver Figura 6.16).

Ahora se lo analiza desde la perspectiva que el GTIEX' realice las estrategias necesarias para alinear las estrategias internas de GTIN'' e Infraestructuras de TI (Ver Figura 6.17).

6.7.1. Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN''

En la Figura 6.17 se presenta el proceso de alineamiento 2 y perspectiva (4) con GTIEX'' y GTIN'', la cual debe adaptarse para cumplir los requerimientos del GTIEX''.

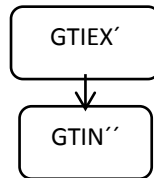


Figura 6.17 Proceso de Alineamiento (2) Perspectiva (4) GTIEX' - GTIN''

En la perspectiva GTIEX' - GTIN'', GTIEX' ya cambió su estrategia de entrega de servicios externos y por tanto GTIN'' debe acoplarse a los nuevos requerimientos de GTIEX' (Ver Figura 6.18).

- *GTIN04: Asignación adecuada de recursos de TI en base a requerimientos para Provisión de Servicios mediante Dispositivos Móviles.*
- *Resultado: GTIN' se actualiza y se convierte en GTIN''*
- *RGTIN03: Definición de tácticas para automatización de procesos para provisión de servicios mediante dispositivos móviles*
- *SEGTIN05: Provisión de componentes de infraestructura de software, computacionales, mecánicos y eléctricos para entrega de prestaciones mediante dispositivos móviles.*

Figura 6.18 Establecimiento de nuevas estrategias luego de analizar GTIEX' sobre GTIN''

El conjunto de una nueva estrategia, requerimiento y servicio, surge como resultado de la ejecución de GTIEX' como cabeza en la modernización de los servicios relacionados con la reconexión de energía eléctrica, en tiempos mínimos luego del corte, como consecuencia del apoyo de dispositivos móviles.

GTIEX' y GTIN'' con sus recursos de gestión de sistemas informáticos e infraestructura de TI, y su relación directa con el usuario interno, encuentran que sus dispositivos móviles pueden ser usados para envíos de mensajes oportunos mientras se encuentran en los lugares asignados para realizar sus labores de corte de energía eléctrica.

Personal de operadores del área de cortes de energía manifiesta que este requerimiento ya había sido entregado al personal de TI, sin embargo no se había tomado en cuenta como una necesidad imperante.

TI responde que no tenía facultades administrativas para cambiar funciones de los operadores de corte.

Este juego de palabras y responsabilidades impidió que se desarrollen con anterioridad aplicaciones estratégicas para todos los involucrados, el usuario interno, cliente externo, organización ejecutiva y TI.

Al estar GTIEX' al frente de la modernización corporativa, tiene la facultad de sugerir cambios estratégicos, a fin de dar solución inmediata a los problemas planteados.

GTIEX' tiene su apoyo en GTIN'' en la definición de recursos que TI debe proveer para realizar software en base a dispositivos móviles.

La Tabla 6-15 presenta la nueva estrategia, requerimiento y servicios que se presentan como adaptación del GTIEX' sobre GTIN''.

Tabla 6-15 Servicios y requerimientos nuevos luego de análisis de GTIEX' sobre GTIN''

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
GTIEX'	GTIEX01 GTIEX02 GTIEX03	RGTIEX'	RGTIEX01 RGTIEX02 RGTIEX03	SEGTIEX'	SEGTIEX01 SEGTIEX02 SEGTIEX03 SEGTIEX04
GTIN'''	GTIN01 GTIN02 GTIN03 GTIN04	RG TIN'	RGTIN01 RGTIN02 RGTIN03	SEGTIN'	SEGTIN01 SEGTIN02 SEGTIN03 SEGTIN04 SEGTIN05

6.7.2. Proceso de Alineamiento (2) Perspectiva (4) GTIN''' - PIN''

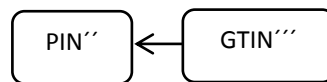


Figura 6.19 Proceso de Alineamiento (2) Perspectiva (4) GTIN''' - PIN''

La figura anterior muestra la perspectiva GTIN''' - PIN'', GTIN''' ya cambió su estrategia, servicios y requerimientos por lo tanto PIN'' debe acoplarse a los nuevos requerimientos de GTIN''' (Ver Figura 6.19 y Figura 6.20).

- *PIN04: Modernización de prestaciones de reconexión de energía eléctrica, que permita fluidez en el servicio.*
- *Resultado: PIN'' se actualiza y se convierte en PIN'''*
- *RPIN04: Desarrollo de aplicaciones para modernizar los Servicios de Reconexión de Servicio de Energía eléctrica mediante dispositivos móviles*
- *SEPIN05: Provisión de componentes de infraestructura administrativa para entrega de prestaciones mediante dispositivos móviles.*

Figura 6.20 Establecimiento de nuevas estrategias luego de analizar GTIN''' sobre PIN''

La relación de GTIN''' sobre PIN'' ayuda a establecer la necesidad de uso de los dispositivos móviles del usuario interno, y para ello necesita contar con los recursos de infraestructura de TI y los recursos de infraestructura administrativa con procesos optimizados y claramente definidos en relación al uso de dispositivos móviles por parte de los operadores de corte de energía.

GTIN''' cumple la necesidad de los requerimientos de GTIEX' y este a PEX, en cadena al cliente externo.

En la Tabla 6-16 se presentan los nuevos servicios y requerimientos fruto del análisis del GTIN''' sobre PIN''.

Tabla 6-16 Nuevos servicios y requerimientos luego de analizar GTIN''' sobre PIN''

Estrategia del Negocio					
Modernización del servicio de reconexión de energía eléctrica					
GTIN'''	GTIN01 GTIN02 GTIN03 GTIN04	SEGTIN'	SEGTIN01 SEGTIN02 SEGTIN03 SEGTIN04 SEGTIN05	RGTIN'	RGTIN01 RGTIN02 RGTN03
PIN'''	PIN01 PIN02 PIN03 PIN04	SEPIN''	SEPIN01 SEPIN02 SEPIN03 SEPIN04 SEPIN05 SEPIN06	RPIN''	RPIN01 RPIN02 RPIN03 RPIN04

6.7.3. Análisis de Requerimientos alineados entre Dominios de SAM

Con el fin de dar solución al eje estratégico del PMESEE: “Modernización del servicio de reconexión de energía eléctrica”, se plantean los requerimientos expuestos en la Tabla 6-17, Tabla 6-18, Tabla 6-19, Tabla 6-20, como resultado de los procesos de alineamiento 1 y 2 con sus respectivas perspectivas (1)-(2)-(3)-(4).

La Tabla 6-17 permite verificar que existe el requerimiento de dar fluidez a la entrega del Servicio de energía eléctrica, y se solicita que los procesos relacionados se optimicen. Al realizar la relación entre dominios, se pudo establecer que la reconexión del servicio de energía eléctrica denota claramente la necesidad de establecer procesos que permitan continuidad en el servicio de energía eléctrica.

Tabla 6-17 Resumen Requerimientos Externos Corporativos

Requerimiento	Descripción
RPEX01	<ul style="list-style-type: none"> Determinar pautas para dar fluidez en entrega de Servicio de energía eléctrica al usuario
RPEX02	<ul style="list-style-type: none"> Optimizar procesos para disminuir tiempos de reconexión de servicio de energía eléctrica

En la Tabla 6-18 se aprecia que los Requerimientos Internos de la Planificación Interna ya solicitan la Optimización de procesos relacionados con los servicios de reconexión de energía eléctrica. Adicional se solicita entregar a TI procesos optimizados para el desarrollo de aplicaciones relacionados con las actividades de reconexión de energía eléctrica.

Tabla 6-18 Resumen Requerimientos Internos Corporativos

Requerimiento	Descripción
RPIN01	- Coordinar la Definición y Optimización de procesos relacionados con los servicios de reconexión de energía eléctrica.
RPIN02	- Canalizar acciones de alineamiento entre infraestructura administrativa e Infraestructura de TI con el fin de entregar procesos optimizados para la modernización del proceso de

	reconexión de energía eléctrica
RPIN03	<ul style="list-style-type: none"> • Coordinar definición de Procesos optimizados para Provisión de Servicios mediante Dispositivos Móviles.
RPIN04	<ul style="list-style-type: none"> • Desarrollo de aplicaciones para modernizar los Servicios de Reconexión de Servicio de Energía eléctrica mediante dispositivos móviles.

La Tabla 6-19 relaciona los requerimientos externos de GTI para alinear los recursos de TI con el eje estratégico corporativo y lo estructura en base al requerimiento de la determinación de los diferentes tipos de aplicaciones que pueden desarrollarse a efectos de lograr optimización de tiempos en la reconexión de energía eléctrica.

Tabla 6-19 Consolidación Requerimientos Externos de GTI

Requerimiento	Descripción
RGTIEX01	<ul style="list-style-type: none"> • Determinar los recursos de TI relacionados con la entrega del Servicio de energía eléctrica.
RGTIEX02	<ul style="list-style-type: none"> • Determinar los tipos de aplicaciones que se necesitan para modernizar los Servicios de Reconexión de Servicio de Energía eléctrica.
RGTIEX03	<ul style="list-style-type: none"> • Determinar Procesos Externos e Internos adaptados y optimizados a las nuevas tecnologías móviles que ofrece TI.

En la Tabla 6-20 se visualiza con claridad la necesidad del desarrollo y despliegue de aplicaciones móviles que ayude al eje estratégico de Modernizar el servicio de reconexión de energía eléctrica.

Tabla 6-20 Consolidación de Requerimientos Internos del GTI

Requerimiento	Descripción
RGTIN01	<ul style="list-style-type: none"> • Coordinar acciones para desarrollar y desplegar aplicaciones eficientes que permitan la Modernización del servicio de reconexión de energía eléctrica.
RGTIN02	<ul style="list-style-type: none"> • Coordinar acciones para desarrollar aplicaciones que ayuden a modernizar los procesos optimizados por infraestructura administrativa.
RGTIN03	<ul style="list-style-type: none"> • Definición de tácticas para automatización de procesos para provisión de servicios mediante dispositivos móviles.

RGTIN01, RGTIN02 y RGTIN03, plantean los requerimientos que deben ser atendidos por TI, cuya concepción es: coordinar acciones y tácticas para desarrollar y desplegar aplicaciones eficientes en el marco de procesos relacionados optimizados, para la provisión de prestaciones mediante dispositivos móviles.

La condición para dar paso al cumplimiento de los requerimientos es que la infraestructura administrativa como son los procesos relacionados con los requerimientos, se encuentren optimizados y que la provisión de componentes de infraestructura de TI se encuentre identificada y disponible.

El equipo DevOps plantea el flujo de tareas más adecuado para dar solución a los requerimientos de la Tabla 6-20.

El flujo de tareas de la Tabla 6-21 es definida entre el equipo DevOps y el usuario.

Tabla 6-21 Flujo de tareas para el equipo de DevOps

Código Tarea (CT)	Descripción de Tarea
1	Realizar App para envío al cliente de recordatorio de pago vía mensaje a celular y correo electrónico, 2 días antes de la fecha de vencimiento del pago de la factura.
2	Realizar app para visualización en dispositivo móvil (Android) del operador la información de cortes del día
3	Realizar app para actualizar en la base de datos vía app de celular del corte realizado
4	Realizar app para emitir mensaje automático por parte del sistema al celular del operador de reconexión más cercano al sitio
5	Realizar app para que Operador actualice estado de reconexión realizada en su celular

De esta manera se logra determinar las tablas con requerimientos estratégicos a ser automatizados, con sus respectivos servicios alineados por dominio.

Aquí finaliza la tarea de SAM-RS, los requerimientos y servicios han sido definidos y permiten superar la zona oscura, por lo tanto se procede con el proceso de desarrollo de software respectivo.

6.8. Desarrollo de aplicaciones con herramientas DevOps en base a requerimientos definidos

En esta sección se realizan las siguientes actividades descritas en la hoja de ruta correspondiente para planificar estrategias para enfrentar automatización de requerimientos.

- Planificar roles y ciclo de vida DevOps
 - Definir equipo de trabajo
 - Líder de equipo
 - Desarrollo
 - Operadores
 - Usuario
 - DevOps
 - Planificar tareas a automatizar
 - Organizar distribución de tareas
 - Planificar desarrollo de la Iteración
 - Roles y responsabilidades
 - Herramientas DevOps a utilizar
 - Desarrollo: Integración continua
 - Despliegue: Configuración
 - Control de Versiones
 - Control de Monitoreo
 - Métricas a utilizar (tiempo de ciclo)
 - Planificar pruebas de la iteración
 - Planificar ciclo de vida del despliegue de la iteración

El desarrollo de la aplicación se lo expresa en términos del flujo de tareas a ser automatizadas en base al requerimiento central planteado, expresado en la Tabla 6-21.

Por razones de políticas de confidencialidad no es posible mostrar el código fuente desarrollado. Se ha definido un pseudocódigo muy cercano al código fuente, con las precisiones oportunas de aplicación de los conceptos y herramientas DevOps utilizadas en los aplicativos.

6.8.1.1. Definición de Metodología DevOps a ser utilizada para el desarrollo y despliegue de aplicaciones

- Equipo de Trabajo-ET
 - Dueño del Producto-DP
 - Líder de Equipo-LE
 - Desarrolladores-DE
 - Operadores-OP
 - Lista del Producto

- Definición de Ciclo de Vida DevOps
 - Planificación de Requerimientos a ser Automatizados
 - Planificación de Priorización Tareas
 - Planificación de la Iteración
 - Desarrollo de la Iteración
 - Pruebas Unitarias
 - Despliegue de la Iteración
 - Control en Producción

- Dueño del Producto
 - El personal encargado de llevar adelante el proyecto es el Dpto. de Análisis Administrativa de la dirección comercial.
 - Número de personas asignadas al proyecto 1

6.8.1.2. Equipo de trabajo

Asignación total de programadores para atender solicitud de programación:

- 1 Líder de Equipo
- 4 programadores
- 1 operador de data center

6.8.1.3. Planificación de Requerimientos a ser Automatizados.

En la Tabla 6-21 se presenta el flujo de tareas a ser automatizados como resultado de los requerimientos RGTIN01, RGTIN02 y RGTIN03.

6.8.1.4. Análisis y Definición de tareas a ser automatizadas.

La definición de tareas es responsabilidad del equipo de desarrollo. No se incluyen en el análisis de SAM-RS, ya que es una tarea netamente operativa relacionada directamente con el análisis de tareas a automatizar.

Responsables: Equipo Desarrollador

- Enviar al usuario recordatorio vía mensaje a celular, correo electrónico, 2 días antes de la fecha de vencimiento del pago de la factura.
- Enviar mensaje al usuario con aviso de corte del servicio de energía eléctrica
- Disponer en dispositivo móvil del operador el listado de cortes del día
- Realización de corte de energía por parte del operador
- Ingresar al sistema vía app de celular del corte realizado
- Emitir mensaje automático por parte del sistema al celular del operador de reconexión más cercano al sitio
- Realizar la reconexión del servicio
- Operador actualiza estado de reconexión realizada a través de app diseñada para el efecto en su celular
- Se actualiza en la base de datos el nuevo estado

6.8.1.5. Planificación de Priorización Tareas

Responsables: Equipo DevOps

La priorización de tareas se la realiza en base a la Tabla 6-21. Una vez definidas se procede a la planificación de la iteración por parte del equipo DevOps.

6.8.1.6. Tiempo de entrega del proyecto con las 5 tareas: 9 semanas.

La Planificación de las Iteraciones para las tareas se encuentra establecida de la siguiente manera:

- Organización del Equipo por tarea:
- *Métricas a ser usadas: Tiempo de Ciclo*
- La Tabla 6-22 presenta la manera de enfrentar el desarrollo de las aplicaciones por tarea. Los recursos humanos, desarrolladores, operadores, dueño del producto por cada tarea, tiempo de ciclo que es el límite de tiempo asignado en días para cada tarea.
- A los 5 miembros del equipo DevOps se los ha detallado como: A, B, C, D, E.
- Equipo de desarrolladores es A, B, C, D
- Operador E
- Al líder del equipo DevOps se lo ha definido como F.

Tabla 6-22 Tiempos y frecuencia de tareas por responsable

Organización Equipo Trabajo	Líder Equipo (LE)	Dev	Ops	Dueño Producto (DP)	Tiempo Ciclo (días) (TC)
Tarea 1	1 F	2 A-B	1 E	1	15
Tarea 2	1 F	1 C	1 E	1	15
Tarea 3	1 F	1 D	1 E	1	15
Tarea 4	1 F	1 A	1 E	1	15
Tarea 5	1 F	1 B	1 E	1	15

- Software de Programación Java
- Base de Datos: Oracle V11g
- Repositorio de Versionamiento: Local en Data Center de la Institución
- Control de Versionamiento: Git
- Repositorio de Almacenamiento de Versiones: GitHub
- Software para despliegue de máquinas virtuales: Vagrant, VirtualBox
- Software para Integración Continua: Jenkins
- Uso de Puppet, para configurar y desplegar software Centos, Java, Mysql en el servidor
- Uso de Nagios para monitoreo de servicios, tanto de sistema operativo como de base de datos.

Herramientas DevOps a ser usadas: Las herramientas por tarea que son usadas se presentan en la Tabla 6-23.

Tabla 6-23 Herramientas DevOps usadas en las aplicaciones por tarea

Organización Equipo Trabajo	Java	Oracle	Git	GitHub	Vagrant- Virtual Box	Jenkins	Puppet	Nagios
Tarea 1	Si	Si	Si	Si	Si	Si	Si	Si
Tarea 2	Si	Si	Si	Si	Si	No	Si	Si
Tarea 3	Si	Si	Si	Si	Si	No	Si	Si
Tarea 4	Si	Si	Si	Si	Si	No	Si	Si
Tarea 5	Si	Si	Si	Si	Si	No	Si	Si

- Operador realizó análisis de espacio disco, memoria, para todas las aplicaciones.

6.8.1.7. Planificación de tareas en 9 semanas de las 5 tareas. (Ver Figura 6.21).

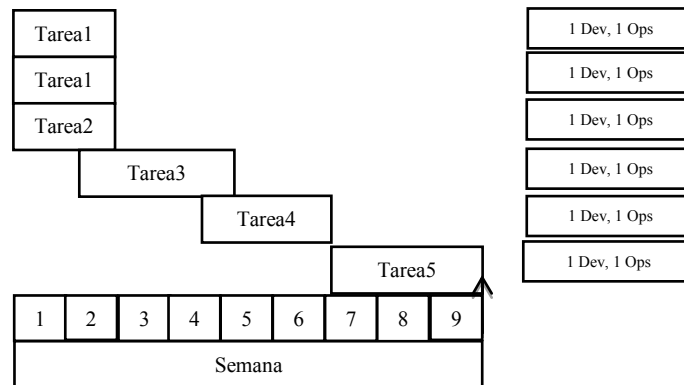


Figura 6.21 Cronograma de actividades por tareas

6.8.1.8. Planificación de Iteración tarea 1

Realizar App para envío al usuario de recordatorio de pago, vía mensaje a celular y correo electrónico, 2 días antes de la fecha de vencimiento del pago de la factura.

Programador 1: Realizar App para envío de mensaje al celular del usuario de recordatorio de pago.

Programador 2: Realizar App para envío de mensaje al correo electrónico del usuario de recordatorio de pago.

Actividades Desarrollador 1 (Dev1) y Desarrollador 2 (Dev2). Aplicación con principios de DevOps: Integración Continua

Se realizó por parte de Dev1 la App para envío al usuario de recordatorio de pago vía mensaje a celular del usuario, de acuerdo al siguiente flujo de actividades:

- Se planificó la iteración a tiempo de ciclo de 15 días,
 - Significa que cada 15 días se debe desplegar el aplicativo para que interactúe el usuario
- Inicio 1er día: Dev1, Dev2 y Ops
 - Dev1: Desarrollo de pruebas de envío de mensajes a celular mediante pruebas de líneas de comandos de sistema operativo.
 - Dev2: Desarrollo de pruebas de envío de mensajes a correo electrónico mediante pruebas de líneas de comandos de sistema operativo.
 - Análisis de las pruebas.
 - Con pruebas satisfactorias se empieza el 2do día.
- Inicio 2do día: Dev1, Dev2 y Ops
 - Dev1: Desarrollo de aplicativo para envío de mensajes a celular.

- Protocolo a utilizar SMTP
 - Dev2: Desarrollo de aplicativo para envío de mensajes a correo electrónico.
 - Integración continua con Jenkins
 - Parte de código de Dev1 y Dev2 controla cada Desarrollador
 - Pruebas con Ops envío de mensajes a través de protocolo SMTP insertado en java.
 - Ops entregan servidores para los aplicativos y rutas para encaminar los mensajes sea a celular o correo electrónico.
- Inicio 3er día: Dev1 y Dev2, Ops1 y Usuario
 - Secuencia de actividades a realizadas:
 - Programa revisa cada día las fechas de vencimiento de la respectiva tabla y suma dos días para comparar con la fecha de los siguientes dos días y la fecha de vencimiento de la factura.
 - El programa debe sumar dos días a la fecha actual y comparar con la fecha de vencimiento de la tabla factura
 - A los clientes que tengan dos días para fecha de vencimiento hay que enviarles el siguiente mensaje a su celular y a su correo electrónico
 “Estimado cliente evite molestias de corte de energía eléctrica, pague a tiempo su factura”
 - Operador de data center pre-configuró ambientes de servidor donde va a residir la aplicación
 - Operador de data center pre-configuró ambiente con Vagrant para virtualización con VMware de máquinas virtuales
 - Despliegue al usuario para que interactúe con el aplicativo
- Inicio 4to día: Dev1, Dev2, Ops1, Usuario
 - Se realizan pruebas con el fin de verificar que llegue el mensaje y que el texto esté acorde al contenido de texto que se desea enviar al usuario
 - Operador de data center pre-configura ambientes de servidor donde va a residir la aplicación
 - Operador de data center pre-configuró ambiente con Vagrant para virtualización con VMware de máquinas virtuales
 - Dev1, Dev2, Ops1, Usuario, verifican que resultado de las pruebas de envío de mensajes vía celular y correo electrónico sea exitoso.
 - Se despliega al usuario los cambios en la aplicación para envío de mensajes masivos.
 - Se verifica funcionamiento satisfactorio.
 - Se coordina para que el Dpto. de Análisis Administrativa de la dirección comercial y el Dpto. Administrativo y soporte de TI realicen la contratación de una base celular para envío de mensajes.

- Inicio 5to día

- Se realizó por parte de Dev1 y Dev2 la activación de constraints a nivel de tablas.
- Se definió formato y estándar de presentación visual.
- Se definió el formato y estándar de control de versiones.
- Inicio 6to día: Dev1, Dev2, Ops1
 - Despliegue en servidor
 - Se realizó pruebas visuales con el usuario.
 - Se realizó pruebas de funcionamiento con usuario.

Tiempo real que tomó la Iteración 1: 6 días (Ver Figura 6.22)

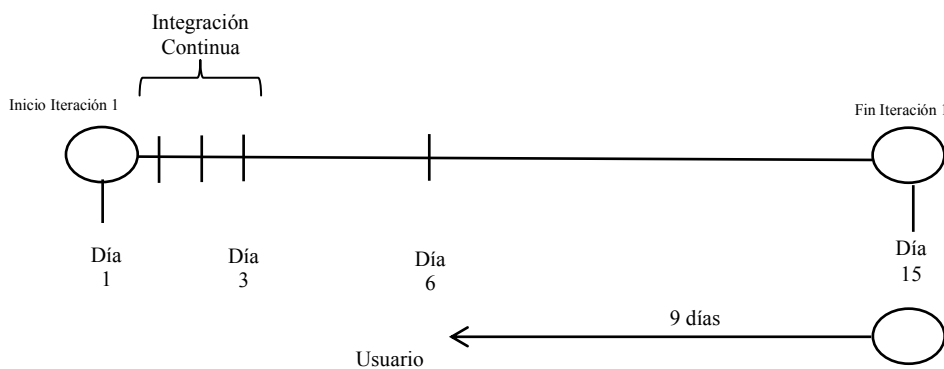


Figura 6.22 Tiempos de Iteración 1

6.8.1.9. Flujo de actividades desarrolladas para la Tarea 2

- Realizar app para visualización en dispositivo móvil (Android) del operador la información de cortes del día

Desarrollador 3 (Dev3)

- Se dividió la iteración en tiempos de ciclo de 10 días.
- Inicio 1er día: Dev3, Ops2
 - Realizar sql que identifique en la base de datos de producción los usuarios que se encuentran morosos al día
 - Realizar una app con archivo cron (creado con Puppet) que se ejecute a una hora determinada entre las 01:00 y 05:00 con el fin de no molestar el tráfico de red a usuarios y que la base de datos se encuentre actualizada con los usuarios que se encuentren en estado de no pago
 - Pruebas de la app Dev3, Ops2
- Inicio 2do día: Dev3, Ops2

- Realizar una app para el dispositivo móvil del operador que se actualice al momento de ejecutar la aplicación con la información que le corresponde a dicho operador con los cortes a realizar en el día.
- Pruebas de la app
- Error de la aplicación. No se actualizan los datos
- Verificación de error
- Pruebas de la app
- Verificación de la app entre Dev y Ops.
- Funcionamiento de la app satisfactoria

El flujo de la iteración 2 sigue como se muestra en la Figura 6.23.

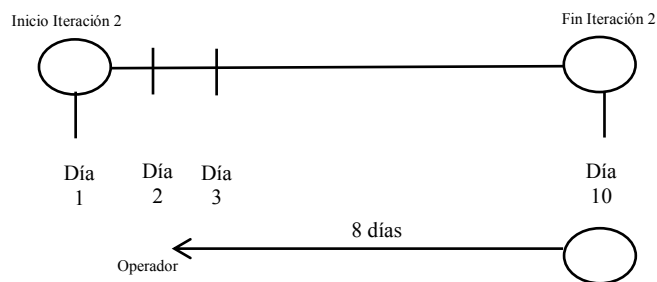


Figura 6.23 Tiempos de Iteración 2

6.8.1.10. Flujo de actividades desarrolladas para la Tarea 3

- Realizar app para actualizar el corte realizado en la base de datos vía app que se encuentra instalada en el celular.

Desarrollador 4 (Dev4)

- Se dividió la iteración en tiempos de ciclo de 8 días.
- Inicio 1er día: Dev4, Ops1
 - Se inicia el desarrollo del app para actualizar el corte de energía eléctrica.
 - Realizar app para insertar en la tabla cortes_reconexión de la base de datos, mediante el celular, el corte realizado.
 - El campo código_cliente es enviado como parámetro para el respectivo corte del servicio.
 - Enviar mensaje al celular y correo electrónico al cliente (con parámetro de código_cliente) que su servicio de energía eléctrica ha sido desconectado por falta de pago. (Ver campos en la Figura 6.24).

<p>TIC-ESEE SISTEMA DE CORTES Y RECONEXIÓN</p> <p>Código Cliente: Nombre Cliente: Dirección: Valor en mora: Código Operador: Nombre Operador: Fecha Corte: Hora de corte:</p>

Figura 6.24 Campos de la aplicación

- Inicio 2do día: Dev4
 - Dev4: Desarrolla aplicación
- Inicio 3er día: Dev4
 - Dev4: Desarrolla aplicación
- Inicio 4to día: Dev4, Ops1, Operador de corte
 - Se realiza pruebas de actualización de datos
 - Fallo en la aplicación. No conecta a base de datos
 - Revisión de aplicación
- Inicio 5to día: Dev4, Ops2, Operador de corte
 - Revisión de aplicación por Dev4
 - Pruebas de la aplicación
 - Aplicación funcionando
 - Pruebas con operador de corte
- Inicio 6to día: Dev4, Ops2, Operador de corte
 - Despliegue de aplicación

Para mejor visualización el flujo de actividades de la iteración 3 se presenta en la Figura 6.25.

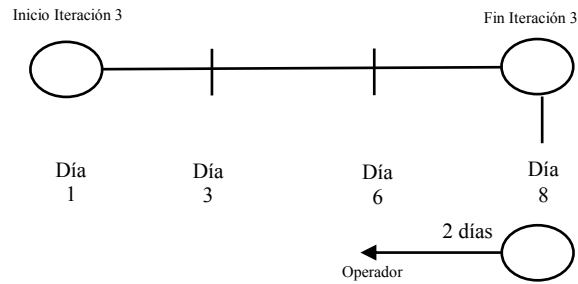


Figura 6.25 Tiempos de la Iteración 3

6.8.1.11. Flujo de actividades desarrolladas para la Tarea 4

- Una vez que el usuario paga la factura respectiva y con el fin de que el servicio de energía eléctrica que le han cortado por falta de pago sea reestablecida en forma inmediata. Se necesita:
Realizar app para emitir mensaje automático de pago realizado por parte del sistema al celular del operador de reconexión más cercano al sitio

Desarrollador 1 (Dev1)

- Se dividió la iteración en tiempos de ciclo de 8 días
- Inicio 1er día: Dev1, Ops1
 - Se inicia el desarrollo de la app para enviar mensaje una vez realizado el pago respectivo
 - Modificar el programa de recaudación:
 - Cuando se realiza el pago respectivo de la factura por parte del cliente, se ingresa el pago a la tabla recaudación y luego de hacer commit de la transacción el programa debe realizar lo siguiente:
 - Comparar código_cliente de la tabla recaudación con código_cliente de la tabla cortes_reconexión.
 - Revisar estado: si el estado está en C
 - Enviar el siguiente mensaje al operador del corte de ese registro de la tabla cortes_reconexión
Mensaje "Factura cancelada favor reconectar en forma inmediata servicio de energía eléctrica al cliente": código cliente y nombre del cliente (de la tabla cortes_reconexión)

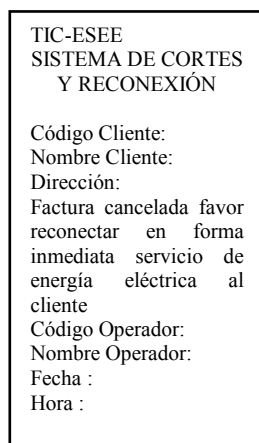


Figura 6.26 Mensaje de facturada cancelada directo al usuario de la aplicación

En la Figura 6.26 se visualiza el mensaje que le llega al operador una vez que ha sido cancelada la factura por parte del cliente. De esta manera el operador puede tomar acciones inmediatas en el sitio.

- Inicio 2do día: Dev1, Ops1, Operador de cortes
 - Realizar pruebas con operador de cortes
 - Pruebas funcionales correctas
 - Despliegue de aplicación

La Figura 6.27 presenta la estructura de los tiempos planificados y tiempo real de la iteración 4.

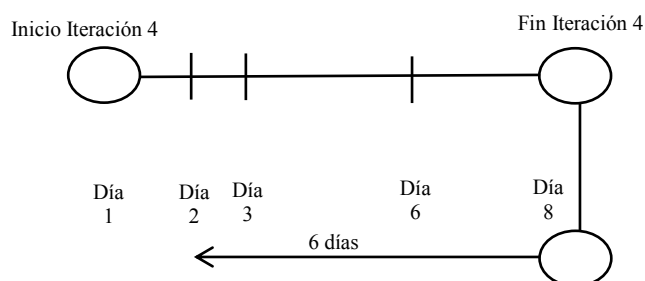


Figura 6.27 Tiempos de Iteración 4

6.8.1.12. Flujo de actividades desarrolladas para la Tarea 5

- Realizar app para que operador actualice nuevo estado de reconexión desde su celular en la tabla cortes_reconexiones y emitir mensaje automático de pago realizado por parte del sistema al celular y correo electrónico del usuario

Desarrollador 1 (Dev1)

- Se dividió la iteración en tiempos de ciclo de 6 días
- Inicio 1er día: Dev1, Ops1, Usuario

- Se inicia el desarrollo de la app para actualizar estado de reconexión
- Realizar pruebas con usuario
- Despliegue de aplicación

La Figura 6.28 presenta la estructura de los tiempos planificados y tiempo real de la iteración 4.

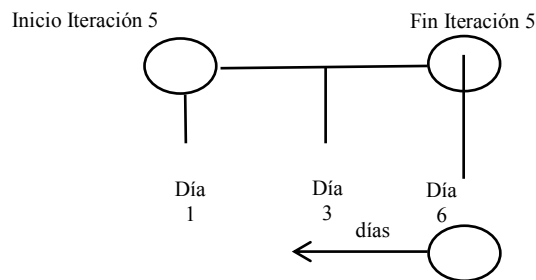


Figura 6.28 Tiempos de Iteración 5

En la figura anterior se puede notar claramente como disminuyen los tiempos al finalizar la tarea con respecto a los tiempos planificados.

6.8.2. Evaluación del modelo de alineamiento SAM-RS.

Para evaluar el modelo SAM-RS se utiliza la métrica DevOps Net Promoter basada en una encuesta realizada según el siguiente flujo de procesos (Ver Tabla 6-24). Se evalúa a SAM-RS y no a la hoja de ruta del modelo orientado a la administración eficiente de recursos de un data center en razón que al momento de utilizar SAM-RS intervienen los 12 participantes del proyecto y pueden dar un criterio más acertado de la efectividad del SAM-RS. En la hoja de ruta no intervienen las 12 personas, por lo tanto no es posible evaluarla con certeza.

- Realizar una encuesta a los 12 participantes, para saber SAM-RS cumplió con las expectativas. Para saberlo se lo determina con la variable de recomendación del uso del modelo a otras personas.
- Para las dos primeras preguntas se tiene las siguiente opciones:
 - 1 Bajo
 - 2 Mediano
 - 3 Alto
- Para la tercera pregunta se tiene las siguientes opciones:
 - 1 Nada
 - 2 Mediano
 - 3 Alto
- Para la cuarta pregunta se plantea las siguientes posibilidades:
 - Si, si recomendaría
 - No, no recomendaría

Tabla 6-24 Encuesta para evaluar Plan estratégico y modelo SAM-RS

Personal (Administrativo 1-6) (TIC 7-12)	(1) Grado de Entendimiento del PE y PEGTI	(2) Grado de Entendimiento del modelo SAM-RS	(3) Resultó ser Efectivo el modelo SAM-RS?	(4) Recomendaría el uso de SAM-RS?
(1)	2	2	3	SI
(2)	1	1	2	NO
(3)	3	3	3	SI
(4)	1	2	3	SI
(5)	1	2	3	SI
(6)	1	3	2	SI
(7)	1	2	2	SI
(8)	2	3	3	SI
(9)	1	2	3	SI
(10)	2	2	3	SI
(11)	1	3	3	SI
(12)	1	3	3	NO

6.8.3. Análisis de encuesta para evaluación del modelo de alineamiento SAM-RS.

(1) El Grado de entendimiento del PE y PEGTI presenta los siguientes resultados:

(1)-(5):

- Personal administrativo con grado de entendimiento bajo de su plan estratégico: 4
- Personal administrativo con grado de entendimiento mediano de su plan estratégico: 1
- Personal administrativo con grado de entendimiento alto de su plan estratégico: 1

(6)-(12):

- Personal de TIC con grado de entendimiento bajo de su plan estratégico: 4
- Personal de TIC con grado de entendimiento mediano de su plan estratégico: 2
- Personal de TIC con grado de entendimiento alto de su plan estratégico: 0

Se desprende las razones por las cuales al inicio del proyecto hubo problemas al recopilar los requerimientos de la organización. No se tenía una real comprensión del plan estratégico y TI por parte del personal que llevaba adelante la modernización corporativa.

Este punto de la encuesta fue realizado al mes de iniciado el proceso de modernización con SAM-RS.

Se capacitó al personal para poder enfrentar con mejor argumentos la planificación de requerimientos y servicios que ayuden a cambios efectivos en la organización.

(2) Grado de entendimiento del modelo SAM-RS

(1)-(5):

- Personal administrativo con grado de entendimiento bajo del modelo SAM-RS: 1

- Personal administrativo con grado de entendimiento mediano del modelo SAM-RS: 3
- Personal administrativo con grado de entendimiento alto del modelo SAM-RS: 2

Se desprende que la mitad del personal administrativo tuvo problemas al realizar la entrega de requerimientos con un enfoque estratégico. La otra mitad si entendió SAM-RS y pudo determinar los requerimientos y servicios que presentaban valor agregado a la organización.

(6)-(12):

- Personal de TIC con grado de entendimiento bajo del modelo SAM-RS: 0
- Personal de TIC con grado de entendimiento mediano del modelo SAM-RS: 2
- Personal de TIC con grado de entendimiento alto del modelo SAM-RS: 4

Para el personal de TIC resultó ser más efectivo el grado de entendimiento del modelo SAM-RS ya que se encuentran familiarizados con la recopilación de necesidades.

(3) Resultó ser efectivo el modelo SAM-RS

(1)-(5):

- Personal administrativo que considera nada efectivo el modelo SAM-RS: 0
- Personal administrativo que considera medianamente efectivo el modelo SAM-RS: 2
- Personal administrativo que considera altamente efectivo el modelo SAM-RS: 4

El análisis de efectividad se entiende desde la perspectiva que el personal administrativo que es representante del ámbito de conformidad pudo entender que los ajustes estratégicos y funcionales del modelo SAM-RS, logran determinar con mayor facilidad los requerimientos y servicios, para luego adecuarlos en tareas y prioridades.

(6)-(12):

- Personal de TIC que considera nada efectivo el modelo SAM-RS: 0
- Personal de TIC que considera medianamente efectivo el modelo SAM-RS: 2
- Personal de TIC que considera altamente efectivo el modelo SAM-RS: 4

El personal de TIC considera efectivo el modelo SAM-RS debido a que disminuye la zona oscura que es el conflicto de requerimientos. Para TIC resulta estratégico estar involucrado con las estrategias internas y externas de la organización.

4) Recomendaría el uso de SAM-RS?

(1)-(5):

- Personal administrativo que considera recomendar el modelo SAM-RS: 5
- Personal administrativo que considera no recomendar el modelo SAM-RS: 1

Existe satisfacción por haber conseguido que el modelo SAM-RS pueda ser recomendado (Métrica 2), por la mayoría del personal administrativo representante de la alta dirección.

(6)-(12):

- Personal de TIC que considera recomendar el modelo SAM-RS: 5 (Métrica 2).
- Personal de TIC que considera no recomendar el modelo SAM-RS: 1

En este punto el análisis se centra en determinar las causas que llevan a sugerir que no recomendaría el uso de SAM-RS, una persona que tuvo un alto grado de entendimiento del modelo SAM-RS y que manifestó que tenía un alto grado de efectividad.

Al ser consultado por las razones que no recomienda el modelo SAM, manifiesta que no logró entender el PE y GTI. El estilo tradicional de enfrentar los requerimientos del usuario-desarrollador era su gran fortaleza y no quería salir de su status quo.

6.8.4. Evaluación de las herramientas DevOps

Para evaluar el uso de las herramientas DevOps, se presenta la Tabla 6-25, que ayuda a identificar los resultados obtenidos de la aplicación de las herramientas utilizadas en el caso de estudio 1.

A los 5 miembros del equipo DevOps se los ha detallado como: A, B, C, D, E.

Equipo de desarrolladores es A, B, C, D

Operador E

Al líder del equipo DevOps se lo ha definido como F.

Tabla 6-25 . Control del ciclo de vida para el equipo DevOps

Flujo de Tareas realizadas por el equipo DevOps	A	B	C	D	E	F
Definir equipo de trabajo	Si	Si	Si	Si	No	Si
Estructurar el equipo de trabajo y designar un líder de equipo. Incluir al equipo al dueño del producto	Si	Si	Si	Si	No	Si
Definición de Procesos manuales y automatizados con los que interactúa el dueño del producto	Si	Si	Si	Si	No	Si
Planificación y Definición de Tareas a automatizar	Si	Si	Si	Si	No	Si
Planificación y Definición de Tareas Prioritarias. Definición del ciclo y frecuencia de la iteración	Si	Si	Si	Si	Si	Si
Planificación en equipo de trabajo para que exista Integración, Entrega y Despliegue Continuo de las aplicaciones a realizar	Si	Si	Si	Si	Si	Si
Identificación de Componentes de la Gestión de Despliegue en Data Center - Servidor identificado, sistemas operativo de servidor, manejador de base de datos, software de frameworks	Si	Si	Si	Si	Si	Si
Identificación de Herramientas DevOps como apoyo al Desarrollo, Java, Oracle	Si	Si	Si	Si	Si	Si
Identificación de Herramientas DevOps como apoyo al Despliegue continuo de las aplicaciones a realizar. Uso de Jenkins y plugin de GIT y repositorio GitHub para Integración Continua y Control de Versiones.	Si	Si	Si	Si	Si	Si
Uso de Vagrant para despliegue de máquinas virtuales donde van a residir las aplicaciones en servidores.	No	No	No	No	Si	Si
Uso de Chef Server, para configurar y desplegar	Si	Si	No	No	Si	No

software Centos, Java						
Uso de Puppet, para restaurar servicios y configurar y desplegar software Centos, Java.	No	No	No	No	Si	Si
Instalación de Nagios para monitoreo de servicios, tanto de sistema operativo como de base de datos.	No	No	No	No	Si	Si
Uso de monitoreo de Nagios para etapa de producción	Si	Si	Si	Si	Si	Si
Análisis de la capacitación en herramientas y procesos a automatizar	Si	Si	Si	Si	Si	Si
Análisis y planificación del equipo que permita compartir en tiempos cortos ideas innovadoras, procesos, herramientas.	Si	Si	Si	Si	Si	Si
Identificación de conceptos de métricas DevOps a cumplir	Si	Si	Si	Si	Si	Si
Colaboración. Existió relaciones colaborativas que apoyen el avance del trabajo	Si	Si	Si	Si	Si	Si
Tiempo de Ciclo	15	15	15	15	15	15
Fallos luego de una semana de monitoreo de las aplicaciones instaladas	No	No	No	No	No	No
Satisfacción de equipo de trabajo al usar herramientas DevOps.	Si	Si	Si	Si	Si	Si

6.8.5. Conclusiones de la aplicación de herramientas DevOps

La aplicación de las herramientas DevOps por parte del equipo DevOps permite concluir que:

- La identificación de roles y responsabilidades permitió que el equipo DevOps cumpla con todos los aspectos de la planificación de tareas para llevar adelante el desarrollo de las aplicaciones.
- El líder del equipo F, participó en todos los eventos programados, lo cual le dio facilidad para armar el equipo de trabajo DevOps.
- Se pudo observar que los tiempos reales fueron menores a los programados en las tareas de automatización.
- Los miembros del equipo A, B, C, D tenían muy claro su alcance del desarrollo de aplicaciones.
- El miembro del equipo E, supo manejar su espacio de administrador de sistemas en base a su responsabilidad.
- El líder del equipo identificado como E, intervino en todas las tareas, lo que le permitió tener el control de avance de todas las actividades.
- Los equipos C y D al no cumplir con la aplicación de conceptos de integración continua, demoraron más tiempo en desplegar sus aplicaciones.
- Se ha cumplido con los objetivos ya que la sincronización de los equipos de trabajo DevOps, que permitió el Análisis de Fronteras en los Roles de DevOps, compartición, automatización y métricas y ayudaron a establecer en forma clara el factor determinante que tiene la relación de apoyo colaborativo de DevOps en todas las etapas del desarrollo de software, para alcanzar los objetivos corporativos en tiempos cortos.
- Se ha realizado un análisis de los Principios DevOps como Integración Continua, Entrega Continua, Despliegue Continuo, que permite visualizar el paradigma colaborativo en el desarrollo de software, y junto a metodologías ágiles producen resultados eficaces en tiempos cortos y ordenados.

- Se han definido los límites de DevOps que ayudan a establecer desde donde comienza la labor colaborativa del equipo de trabajo y hasta donde llega la responsabilidad luego del despliegue del todo el proyecto. Esto permite tener claro las responsabilidades de recuperación de fallos de las aplicaciones.
- Se ha analizado los alcances del concepto de DevOps, que ayudan a entender los nuevos paradigmas de colaboración entre las áreas de desarrollo y operadores, para todo el proceso de desarrollo de software. De esta manera se logra que cada etapa de la metodología ágil en base a DevOps finalice en forma eficiente en tiempos cortos.

6.8.6. Resultados obtenidos del desarrollo de software del caso de estudio 1

Los resultados obtenidos debido a las estrategias aplicadas en cada dominio del SAM, son presentados para el gobierno corporativo, gobierno de TI, el usuario interno y el cliente externo.

- **Gobierno Corporativo**

En base a la aplicación de los procesos de alineamiento con sus perspectivas del SAM, se han obtenido los siguientes resultados:

- Tabla de requerimientos externos e internos corporativos expuestos en la Tabla 6-17 y Tabla 6-18.
- Consolidación de requerimientos externos e internos de TI expuestos en las Tabla 6-19 y Tabla 6-20.
- A partir de ellas se genera el flujo de tareas para el equipo DevOps expuesta en la Tabla 6-21.

De esta manera se obtienen los requerimientos directos en base a la aplicación de los procesos de alineamiento con sus perspectivas del SAM.

- Cumplimiento del Objetivo del eje corporativo: Modernización del servicio de reconexión de energía eléctrica en base a los siguientes cumplimientos:
 - Se han aplicado herramientas DevOps para el desarrollo de las aplicaciones en tiempos menores a los establecidos.
Las aplicaciones ayudan a dar continuidad al servicio de energía eléctrica, ya que se minimiza el tiempo de corte del servicio de energía eléctrica, logrando recuperación económica inmediata.
- La optimización de procesos, producto de la intervención de los dominios de SAM, ligados a la reconexión del servicio de energía eléctrica, ha logrado un significativo impacto de ahorro de tiempo y costos directos e indirectos en los procesos involucrados.

- En base a los antecedentes expuestos se han habilitado nuevos servicios empresariales (métrica 1): Aplicaciones móviles para disminuir tiempos de reconexión del servicio de energía eléctrica.
- Gobierno de TI
 - Debido a la aplicación del pivotaje de la metodología MLS en SAM-RS, se ha logrado obtener requerimientos efectivos mostrados en la Tabla 6-20 que ayudan a efectivizar el desarrollo de aplicaciones.
 - La aplicación de herramientas DevOps ha logrado tareas colaborativas (métrica 4) que ayudan a disminuir los tiempos de desarrollo y despliegue de los aplicativos ahorrando tiempos y disminuyendo costos operativos.
 - Realizadas las aplicaciones se puede verificar que los tiempos planificados de tareas dan como resultado diferencia de tiempos favorables a los usuarios, a TI y a la organización (Ver Figura 6.29).
 - Los tiempos de ciclo (métrica 5) planificados se han optimizado debido a la aplicación de conceptos de integración continua y al uso de herramientas DevOps en al desarrollo de las aplicaciones solicitadas.
- Usuario interno
 - El usuario interno ha conseguido aplicaciones de software que efectivizan su diaria labor. Puede recibir mensajes de reconexión inmediatamente luego que ha pagado su factura, y puede proceder a reconectar el servicio en corto tiempo. Existe satisfacción del usuario interno (métrica 3).
- Cliente externo
 - El cliente externo se beneficia al tener mínimo tiempo sin energía, debido a rapidez con la que se le reconecta el servicio de energía eléctrica, una vez que ha cancelado su factura. (Satisfacción del cliente externo (métrica 3)).

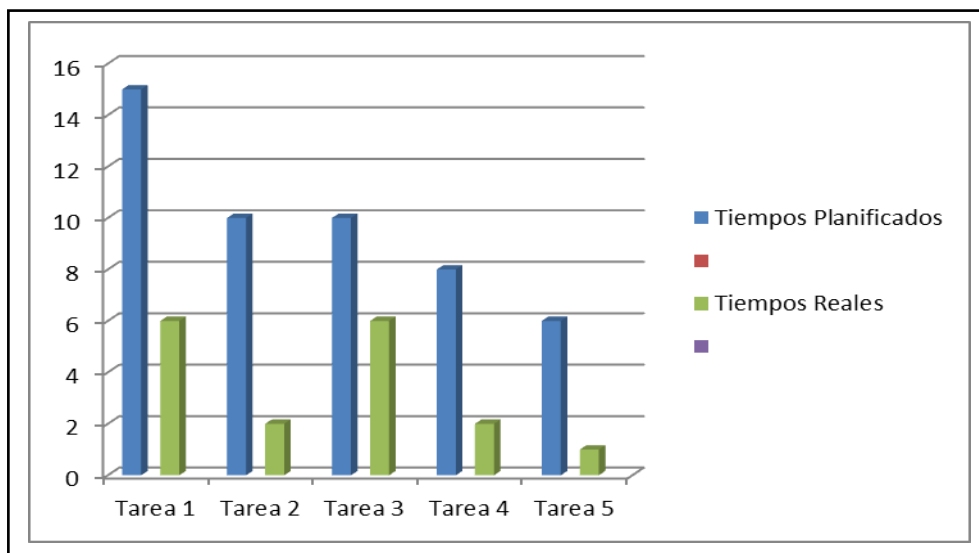


Figura 6.29 Tiempos planificados vs tiempo real

En la figura anterior se visualiza que los tiempos reales con color verde marcan una tendencia favorable al equipo DevOps en relación a los tiempos planificados con color azul.

En este proceso de alineamiento corporativo y TI, existen beneficios directos, sean económicos o de satisfacción de servicios.

En la Tabla 6-26 se puede apreciar que los valores recuperados son importantes para la organización.

Tabla 6-26 Recuperación de valores por disminución en tiempos de reconexión

Cortes Promedio Mensual	Promedio tiempo de reconexión	Valor kw-hora (dólar)	Kw promedio consumidos por hora en sector residencial	Valor por 0,30 horas no consumidas (dólar)	Valor recuperado
29.000	0,30 horas	0,077	1,45	971,35	8742,19

El promedio de tiempo desde la desconexión hasta la reconexión de energía era de 3 horas, ahora es 30 minutos.

El tiempo de 2 horas y 30 minutos, para el cliente externo representa uso de energía en actividades relacionadas con su ámbito diario que no pueden ser fácilmente reemplazadas.

La medición de beneficios para el cliente externo, en las 2 horas y 30 minutos, no es fácilmente cuantificable, y requiere otro tipo de estudio que no es tratado en esta tesis.

No obstante se puede manifestar que son indudables los beneficios que alcanza el cliente externo al tener 2 horas y 30 minutos más de energía para su consumo, a conocer en salud, alimentación, temperatura, trabajo, y se puede manifestar que supera a los \$ 8.742,10 que la organización recuperó por la misma razón.

6.8.7. Conclusión del Caso de Estudio 1

La descripción detallada del flujo de procesos del caso de estudio 1 puede inducir a pensar que esta propuesta no facilita la agilidad y el dinamismo que la organización y el área de TI requieren en sus soluciones.

Inicialmente puede resultar engorroso respetar estas interrelaciones y justificación de las acciones a llevar a cabo por cada dominio y perspectiva. Pero cuando una organización conoce y acompaña los objetivos estratégicos en el marco corporativo, de negocio y TI, enmarcar sus acciones en estos objetivos no es tan complejo.

Se abordan diferentes aristas en la búsqueda de estrategias, requerimientos y servicios en pos de encontrar la mejor alternativa de solución a las necesidades corporativas.

Esta interrelación de dominios, procesos y perspectivas permite visualizar resultados directos y logra establecer el grado de exigencia y viabilidad de los requerimientos solicitados hacia la obtención de objetivos enunciados desde la corporación.

Los problemas planteados, son resueltos debido a la efectividad del alineamiento corporativo y TI mediante la aplicación del SAM-RS y MLS.

La satisfacción del nuevo servicio obtenido se ve reflejada en la organización desde el aspecto económico como en el aspecto de beneficios del usuario interno y cliente externo. (Métricas 1 y 2).

6.9. Caso de Estudio 2

El caso de estudio 2 se propone como una segunda forma de ver la alineación estratégica, desde el gobierno de TI hacia el corporativo, mediante el uso de herramientas DevOps.

Se plantea la aplicación de la herramienta de monitoreo Nagios para realizar un control detectivo de fallos, y luego utilizar la herramienta de configuración Puppet para restaurar el servicio que dejó de funcionar.

Para ello se presenta un modelo de granularidad de los componentes del Despliegue de aplicaciones en el Data Center.

Estas acciones van a permitir un monitoreo contante y minucioso de las aplicaciones cuyos servicios están activos cuando son usadas por los usuarios.

El objetivo es minimizar los errores y controlar la disponibilidad de las prestaciones.

Al lograr un monitoreo preciso se puede restaurar los servicios de los componentes de software que presentan fallos antes que se conviertan en averías.

Las herramientas DevOps aplicadas en componentes granulosos de software que residen en servidores del Data Center, logran alinear los intereses corporativos que brindan servicios a los clientes con efectiva disponibilidad de los recursos y servicios.

6.9.1. Protocolo para control de software en producción

La estrategia para enfrentar el caso de estudio 2 se lo plantea en base al protocolo establecido para el monitoreo de software en producción (Ver sección 4.7.6):

- Determinar el objetivo del monitoreo
- Definir propiedades de apoyo de buenas prácticas apegadas a las necesidades del negocio que pueden ser disponibilidad, continuidad.
- Determinar la herramienta de monitoreo de acuerdo a sus propiedades
- Determinar el rango de tiempo del monitoreo en la barrera de la tolerancia de fallos
- Definir los componentes de software macro relacionados con la aplicación a monitorear
- Detallar los componentes hasta un nivel atómico (que ya no tiene subcomponentes)
- Aplicar el monitoreo sobre el componente definido
- Realizar pruebas de funcionamiento

6.9.1.1. Objetivo del monitoreo

El objetivo del caso de estudio 2 es monitorear los servicios de las aplicaciones en producción y restaurarlos sin intervención del operador en tiempos menores al parámetro de tiempo de espera establecido.

6.9.1.2. Definir propiedades de apoyo de buenas prácticas a las necesidades del negocio

El caso de estudio 2 se soporta en los objetivos, pautas y responsabilidades de las buenas prácticas de ITIL, descritos en la sección 5.2.2:

- Aumentar la confiabilidad de la infraestructura y los servicios.
- Las buenas prácticas ITIL permiten dar pautas para coordinar las acciones que ayuden a tener un control sobre los componentes del Data Center que proveen diferentes servicios.
- Garantizar el nivel de disponibilidad establecido para los servicios de TI, Monitorizar la disponibilidad de los sistemas TI

Para garantizar el nivel de disponibilidad óptimo se identifican las actividades claves de los servicios que ofrecen los distintos niveles de componentes y subcomponentes del Data Center. Las actividades de los servicios de cada componente son identificadas en base a los siguientes flujos de tareas:

- Identificar los componentes por donde atraviesa el flujo informático desde el Data Center hasta el Usuario.
Identificar el nivel de granularidad con que se debe detallar a cada uno de los componentes.
- Detallar cada componente relacionado con los servicios que utiliza el usuario.
- Revisar los controles y monitoreo que se debe aplicar.
- Analizar los tiempos de detección y respuesta a los errores.
- Analizar los tiempos de reparación y recuperación del servicio Para conseguirlo es necesario apoyarse en herramientas de monitoreo y restauración automáticas, sin que tenga que intervenir el personal de operadores.

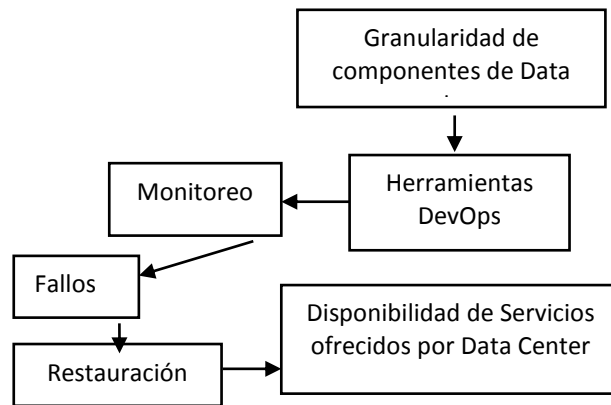


Figura 6.30 Control de monitoreo sobre componentes del Data Center

En la Figura 6.30 se presenta una dinámica donde se aprecia que, la granularidad de los componentes del data center acompañados de un constante monitoreo automático, permiten un adecuado manejo de errores, obteniendo disponibilidad de los servicios ofrecidos al usuario.

6.9.1.3. Determinar la herramienta de monitoreo de acuerdo a sus propiedades

Para contar con la herramienta de monitoreo más adecuada que ayude el objetivo planteado, se tomó en consideración los siguientes aspectos:

- Diferentes métodos de alertas

- Servicios de mensajes, correo electrónico
- Scripts personalizados
- Integración de eventos con el área de mesa de ayuda
- Personalización de la herramienta
- Compatibilidad de sistemas operativos

Para definir los componentes de software macro relacionados con la aplicación a monitorear se detalla algunas consideraciones:

- El aplicativo se encuentra instalado en un servidor con sistema operativo Linux.
- La aplicación utiliza servidor apache.
- Es necesario granular los servicios de apache.
- Monitorear los servicios de apache y restaurarlos en forma automática en caso de fallo.

En base a estos parámetros se han analizado las herramientas de control Nagios y Sensu dando como resultado que son mínimas las diferencias de configuración y resultados al ejecutar comandos que monitorean servicios en los cuales se realizó paradas abruptas. Los tiempos de detección de los servicios caídos fueron similares.

En este estudio se ha preferido optar por Nagios debido a su característica de la Inmediatez tratada en la sección 4.7.5.

Los fallos identificados con Nagios son apoyados por las herramientas DevOps de Configuración que sirven para restaurar, configurar, subir o levantar en forma automática los servicios.

De las herramientas de configuración se ha escogido a CFEngine, Puppet y Chef entre las más utilizadas.

Con el fin de definir cuál de las tres herramientas de configuración es la más adecuada se han realizado las siguientes pruebas:

- Pruebas de desempeño al ejecutar un mismo grupo de instrucciones sobre un mismo servidor.
- Determinar las competencias básicas de cada herramienta
 - CFEngine
 - Desarrollado en C
 - Trabaja en base a dependencias
 - Propio lenguaje declarativo
 - Más opciones de aprendizaje para usuario nuevos
 - Puppet
 - Trabaja en base a Manifiestos
 - Manifiestos utilizan lenguaje Ruby
 - Más fácil de aprender
 - La funcionalidad de Puppet permite: gestión de paquetes, gestión de configuraciones, gestión de scripts y ejecución de comandos, archivos cron, gestión de servicios, gestión de grupos de usuarios.

- Chef
 - Trabaja en base a recetas
 - Recetas utilizan Ruby
 - Tiene opciones como Chef Solo que permiten trabajar en local.
- CFengine es mas estable, sin embargo cuando se lo trabaja con pocos hosts su tiempo es mayor que Chef y Puppet.
- Puppet en cambio es mucho mas rápido que Chef y CFEngine.
- Chef por otro lado, es simple de usar.

Las tres herramientas cumplen funciones y competencias básicas de configuración. rendimiento, configuración de sus componentes, capacidad de mantenimiento de las opciones.

Para este estudio se ha tomado Puppet como herramienta de restauración de los servicios que tienen fallos que han sido detectados con Nagios, debido a su curva de aprendizaje mas pequeña.

Puppet tiene una estructura de Cliente-Servidor, por lo tanto instalamos software tanto en la parte de servidor, como en los clientes.

En el servidor hemos configurado PuppetMaster, que es el que administra los demás nodos y un cliente Puppet en los nodos.

6.9.1.4. Determinar el rango de tiempo del monitoreo en la barrera de la tolerancia de fallos

El parámetro de referencia es el tiempo de espera (timeoutsecs) que es el tiempo máximo de espera antes de abandonar una sesión.

Se encuentra configurado en 5 segundos. Se considera que en este tiempo es posible restaurar servicio de software.

6.9.1.5. Detallar los componentes hasta un nivel atómico (que ya no tiene subcomponentes)

El camino que sigue una aplicación se puede visualizar en la Figura 4.9. Se analizan los componentes de software ya que son los más visibles para reflejar el código que sirve para controlar los servicios activos estableciendo capas por componentes y subcomponentes.

Para la identificación de servicios con riguroso detalle se plantean algunas preguntas:

- Cómo identifico cuando un servicio no está activo?
- Qué tipo de software me ayuda a enviar mensajes cuando un servicio prestado activo tenga un problema?
- El software para monitorear los errores de los servicios tiene interfaz con las diferentes plataformas que necesito monitorear?

Las respuestas a todas estas preguntas se logran identificando y detallando los servicios relacionados desde que el usuario activa una aplicación hasta cuando llega al servidor de aplicaciones y base de datos del Data Center.

Para identificar los servicios de una aplicación que utiliza los servicios del sistema operativo linux, se detallan una secuencia de pasos que van granulando los componentes en concordancia con las aplicaciones del usuario.

En la: Tabla 6-27, Tabla 6-28, Tabla 6-29, Tabla 6-30, Tabla 6-31, se muestra la granularidad de servicios relacionados con una aplicación que utiliza los servicios de Apache.

Tabla 6-27 Procesos relacionados con la entrega del servicio de sistema operativo

SISTEMA OPERATIVO	Servicios activos del sistema operativo
	Servicios de las librerías del sistema operativo para funcionamiento de la base de datos con determinada versión de sistema operativo
	Servicios de las librerías del sistema operativo para funcionamiento de los programas del sistema informático.
SISTEMA OPERATIVO Y BASE DE DATOS COMPARTIENDO SERVICIOS	Servicios del sistema operativo y la base de datos que hacen funcionar a los programas del sistema informático
BASE DE DATOS	Servicios activos de la base de datos
	Servicios de las librerías de la base de datos para funcionamiento de la base de datos con determinada versión de sistema operativo
	Servicios de las librerías de la base de datos para funcionamiento de los programas del sistema informático
SISTEMA OPERATIVO, BASE DE DATOS Y COMUNICACIONES COMPARTIENDO SERVICIOS	Servicios del sistema operativo para conexión del servidor con el switch core del data center
	Servicios de la base de datos para conexión del servidor con el switch core del data center
COMUNICACIONES	Servicios de comunicaciones desde el data center hasta llegar al usuario que incluye conexiones del switch core, tipo de cables utilizados, switch departamentales, tarjetas de red de las computadoras de escritorio, direcciones ip estáticas o dinámicas, sistema operativo del computador de escritorio
	Servicios de tipo de conexión para uso de la aplicación informática vía web o cliente servidor desde el computador de escritorio del usuario hasta llegar al servidor del data center
DESARROLLO	Servicio de mantenimiento de programas fuentes en caso de ser una arquitectura cliente servidor que facilite la instalación de los programas ejecutables
CLIENTE	Servicios del sistema operativo del computador de escritorio del usuario
	Servicios de las librerías que usa la aplicación informática para su funcionamiento en la versión del sistema operativo del computador del usuario
	Servicio de ancho de banda de internet desde el computador del usuario hasta el proveedor para uso de la aplicación web
	Servicio de herramientas de ofimática que se relacionan con la aplicación informática

La relación de los servicios es directa, es decir si falla uno de ellos va a fallar toda la cadena de provisión de servicios.

La Gestión de Control puede darse a este nivel, en la Tabla 6-28 se toman los componentes del Sistema Operativo para visualizarlo:

Tabla 6-28 Procesos relacionados con la entrega del servicio de sistema operativo

SISTEMA OPERATIVO	Servicios activos del sistema operativo
	Servicios de las librerías del sistema operativo para funcionamiento de la base de datos con determinada versión de sistema operativo
	Servicios de las librerías del sistema operativo para funcionamiento de los programas del sistema informático.
Servicios activos del sistema operativo	<ul style="list-style-type: none"> • Cuáles son los Servicios Activos del Sistema Operativo para una determinada versión? • Qué sucede si uno de estos servicios se desactiva o falla? • Cuáles son las razones para que falle? • De qué manera se pueden subir o activar nuevamente el servicio que falló?
Servicios de las librerías del sistema operativo para funcionamiento de la base de datos con determinada versión de sistema operativo	<ul style="list-style-type: none"> • Cuáles son las librerías que se necesitan para el funcionamiento de la base de datos ?
	<ul style="list-style-type: none"> • Qué sucede si una de estas librerías no está completa?
	<ul style="list-style-type: none"> • Se tiene un Plan de Configuración que permita automatizar la actualización de librerías?
	<ul style="list-style-type: none"> • Es necesario cambiar de versión de base de datos o de sistema operativo para que trabajen sin problemas tanto el sistema operativo como la base de datos y qué impacto pueden tener estos cambios?
Servicios de las librerías del sistema operativo para funcionamiento de los programas del sistema informático.	<ul style="list-style-type: none"> • Son compatibles las librerías propias del software con la versión del sistema operativo?
	<ul style="list-style-type: none"> • Se pueden identificar cuáles son los servicios activos del sistema operativo que se relacionan directamente con las librerías que usan los programas del sistema informático

La granularidad de los elementos lleva a la Tabla 6-29:

Tabla 6-29 Servicios activos del sistema operativo

Servicios activos del sistema operativo	<ul style="list-style-type: none"> • Cuáles son los Servicios Activos del Sistema Operativo para una determinada versión? • Qué sucede si uno de estos servicios se desactiva o falla? • Cuáles son las razones para que falle? • De qué manera se pueden subir o activar nuevamente el servicio que falló?
---	---

- Cuáles son los Servicios Activos del Sistema Operativo para una determinada versión?. Esta pregunta se canaliza en la Tabla 6-30.

Tabla 6-30 Servicios activos del sistema operativo Linux ¹⁸

Root	Administrador con control total
Bin	Propietario de las utilidades del sistema operativo.
Daemon	Gestor de servicios generales
adm sys	Propietario de los archivos de registros históricos y administrativos.
Lp	Administrador de los servicios de impresión.
Nobody	Gestor de servicios varios.
ftp	Controlador del acceso al árbol del servicio FTP anónimo..
Sshd	Usuario ficticio gestor del servicio SSH.
Apache www-data	Propietario de los ficheros y directorios del servicio de hipertexto Apache.
Squid Proxy	Controlador del servicio de representación Squid.

En la Tabla 6-31 se definen los servicios activos del aplicativo que está siendo usado por el usuario.

Tabla 6-31 Servicios activos de Apache

<p>Cuáles son los Servicios Activos de Apache para una determinada versión? Están activos los servicios que controla el Servicio de Apache</p> <ul style="list-style-type: none"> - ps Pid <p>Revisión de script .sh para control de servicios</p> <ul style="list-style-type: none"> - status apache2.pid

Cada aplicativo instalado genera archivos y en algunos casos lo hace como librerías y servicios.

Se analiza cada componente y subcomponente para determinar si existe un límite en su granularidad y comprobar si se puede controlar en su punto de fallo.

¹⁸ Tomado de (Ubuntu, 2015)

En el caso de Apache se tiene el servicio httpd el cual es atómico y no tiene subcomponentes. El monitoreo se lo realiza a este nivel con módulos independientes propios de la herramienta de Nagios.

De igual manera sucede con los servicios de bases de datos, sistemas informáticos, tomando en cuenta que se tiene servicios propios de cada componente que están relacionados con otros servicios como los sistemas operativos.

El detalle del sistema operativo se lo debe realizar hasta llegar a un máximo probable de control óptimo de sus elementos con tiempos delta t (Δt) mínimos de restauración de sus servicios.

Habiendo granulado los servicios del sistema operativo hasta un nivel de detalle que nos permita ejecutar comandos que controlen dichos servicios, habremos llegado al límite que nos ayude a resolver el problema en forma automática.

6.9.1.6. Aplicar el monitoreo sobre componente definido

Cada espacio de tiempo resulta estratégico para la detección de errores que pueden ser críticos.

La Gestión de Monitoreo se da a un nivel que permite el control exacto del error del componente. Para ello en cada espacio de tiempo se puede incluir instrucciones con código que verifica si el servicio está activo, al encontrar un error en un tiempo Δt_1 y el monitoreo lo detecta en un tiempo Δt_2 , se ejecuta la orden respectiva con la instrucción que permite restaurar el error producido.

En caso de excepción y no poder levantar el servicio se deberá tener instrucciones que envíen mensajes al operador respectivo para que se lo levante en forma manual. De igual manera en caso de ser válida la restauración se envían mensajes de servicio caído y restaurado en forma correcta. (Ver Figura 6.31).

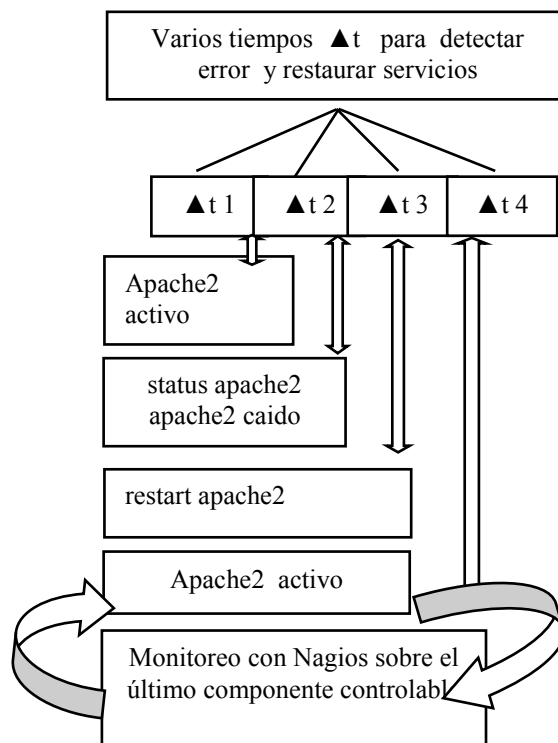


Figura 6.31 Monitoreo de servicios de Apache con Nagios por componentes de sistema operativo

En Nagios se configura el módulo check_apachestatus y se lo ejecuta. Se consideran los tiempos entre la ejecución de la instrucción para verificación del servicio activo como tiempo 0 y la respuesta cuando estuvo caído el valor que corresponde al emitir la señal de fallo. Nos devuelve los siguientes tiempos de respuesta:

Tabla 6-32 Tiempos de duración desde el error hasta su identificación y envío de control

Servicio	Tiempo (seg)
Apache2 activo	0
status apache – apache caído	2
Total	2 (seg)

La Tabla 6-32 presenta el resultado de 2 seg. Significa que en 2 segundos se ha logrado detectar el error.

En la Figura 6.32 aparece el tiempo de recuperación t_r , que nos indica el momento que fue restituido en forma satisfactoria el servicio.

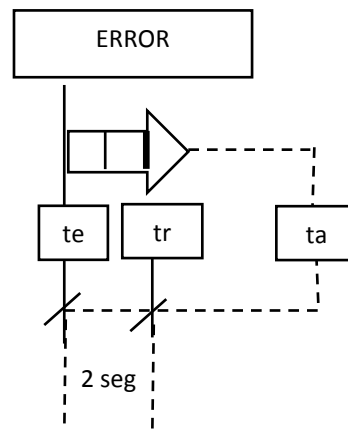


Figura 6.32 Identificación de Tiempos Críticos durante el error

El resultado de la evaluación se presenta en la Figura 6.33:

Si $\Sigma \text{delta } t (\blacktriangle t) > 4 \rightarrow ta > tr$, por lo tanto se llega a tiempo de error y la restauración manual tomará desde que una persona, usuario o mensaje automático, lo notifique hasta que personal de TI ingrese y en forma manual lo vuelva a restaurar.

Si $\Sigma \text{delta } t (\blacktriangle t) < 4 \rightarrow tr < ta$, por lo tanto puede ser restaurado el servicio en forma automática.

Figura 6.33 Resultado de análisis de error

6.9.1.7. Restaurar el servicio con Puppet con mínimos tiempos Δt

El directorio módulos de Puppet contiene el directorio Manifests que indica a los diferentes clientes de Puppet los procesos que se van a llevar a cabo en forma secuencial o de acuerdo como el desarrollador lo programe.

En este directorio se ubica el archivo que contiene el código en ruby necesario para restaurar del servicio.

Al ejecutar el archivo `init.pp` que contiene código (Ver Figura 6.34) se logra restaurar el servicio interno de apache, en tiempos que no son visibles para el usuario y el cliente que está usando la aplicación. Esto se debe a los tiempos permisibles que ofrece el browser que se encuentre activo junto a la aplicación, son superiores a los tiempos que lleva encontrar y restaurar un error con las herramientas DevOps.

```

/manifests/init.pp
    vi init.pp
class delta {
exec { "Restaurar servicio apache2"; command
➔      "/sbin/status          apache2
/sbin/service apache2 restart"
    }
}

```

Figura 6.34 Código de restauración de servicio con Puppet

6.9.1.8. Realizar pruebas de funcionamiento. Resultados obtenidos

En la Tabla 6-33 se presentan los resultados obtenidos:

Tabla 6-33 Tiempos de duración desde la toma de control hasta su restauración con Puppet

Servicio	Tiempo (seg)
Apache2 caído	0
Apache2 activo	0,375
Total	0,375 (seg)

La verificación de tiempos se lo realizó en el siguiente tipo de servidor:

- 1 servidor físico tipo cuchilla o blade marca IBM con las siguientes características:
 - o 4Gb memoria
 - o 300Gb Almacenamiento disco
 - o Procesador 2.8 Ghz
- Sistema operativo Centos 6.2
- Sin carga adicional
- Sin balanceo de carga
- Interfaz de red Ethernet-Gigabit

En la Figura 6.35 se muestran los tiempos de 2,375 segundos, que es el tiempo que ha llevado desde la detección del error hasta su restauración.

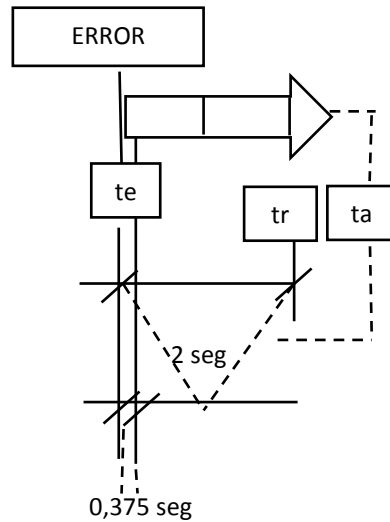


Figura 6.35 Restauración de servicio con Puppet con tiempos críticos luego de un error

La gráfica anterior visualiza que se ha cumplido con el objetivo de monitorear y restaurar un servicio en tiempos de 2,375 segundos, que es menor a los configurados en el *timeoutsec* de 5 segundos.

6.9.1.9. Conclusión de Caso de Estudio 2

En el caso de estudio 2 se analizó la granularidad con la que se deben detallar los componentes de un Data Center de acuerdo a las buenas prácticas de disponibilidad de ITIL con el fin de evaluar los controles adecuados por componente de software.

Si bien es cierto resulta un poco tedioso al inicio la granularidad de niveles y subniveles que se debe dar a los distintos tipos de componentes de un Data Center, sin embargo es muy útil hacerlo y con el uso de herramientas de monitoreo efectivas se puede tener los controles adecuados que permitan fiabilidad de recursos para los usuarios.

La restauración de los servicios en tiempos mínimos se lo ha realizado con herramientas de configuración como Puppet que evita que los operadores realicen esta tarea en forma manual.

El conocimiento de los conceptos de fallo y error es clave para determinar la granularidad que se debe tener en los componentes que dan servicios a los usuarios. De esta forma se puede mantener un control adecuado realizar tareas con mínima intervención de los operadores y restaurar los servicios con el fin de cumplir los acuerdos de niveles de servicio definidos con el usuario.

Se puede afirmar que se ha cumplido varias de las métricas propuestas:

- Habilitación de nuevos servicios empresariales: el monitoreo puede ser una función de las herramientas DevOps. Sin embargo al ser estratégico el servicio de monitoreo que brindan, se lo puede tomar como un nuevo servicio empresarial que es el de mantener alta disponibilidad de los servicios.
- Satisfacción de los usuarios: esta métrica se la asume desde el siguiente punto de vista subjetivo. El aplicativo puede ser estratégico para el usuario, por lo tanto la máxima disponibilidad brindada mediante el monitoreo y

restauración de los servicios relacionados, genera confianza en el usuario interno y cliente externo al recibir productos con eficiencia y eficacia.

6.10. Conclusión de los dos Casos de Estudio

Los casos de estudio explicados en este capítulo muestran el nivel estratégico que tiene el alinear el Gobierno Corporativo y TI.

El alineamiento con el SAM-RS ayuda a establecer estrategias directas entre el Gobierno Corporativo y TI, mediante competencias claves para la identificación de requerimientos y servicios en cada dominio empresarial.

El uso adecuado de las herramientas en el desarrollo y despliegue de aplicaciones con DevOps, ayuda a establecer disponibilidad en las prestaciones brindadas a los clientes y usuarios. Al establecer controles que ayuden a monitorear errores sin llegar a averías, permite mantener niveles de disponibilidad altos sobre las aplicaciones.

El Gobierno Corporativo se siente alineado al Gobierno de TI en base al nivel satisfacción permanente de los clientes, condición de medición clave en los estándares del adecuado manejo de indicadores y recursos en la organización.

7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1. Conclusiones

- Se ha planteado una interfaz entre los componentes de COBIT y SAM a fin de justificar el uso de este, como modelo de alineamiento estratégico.
- Se ha propuesto el modelo de alineamiento estratégico corporativo y TI, SAM-RS, como resultado de la adaptación del modelo SAM en base a requerimientos y servicios.
Se aplicó la metodología *Lean Startup* (Ries, 2016) como apoyo a SAM-RS, a fin de conseguir una cadena implementada mediante sus principios.
Se estructuró SAM-RS en base a requerimientos y servicios para los dos procesos y las cuatro perspectivas, aplicando pivotaje de MLS, a fin de alcanzar la máxima efectividad de requerimientos en base a las distintas maneras de controlar los cambios o modernización corporativa.
Asumiendo que los requerimientos convierten a los Dominios en clientes cuando se realizan requerimientos en cascada, se esperan resultados de acuerdo a las competencias distintivas de cada Dominio.
No es posible entregar valor agregado solamente con las herramientas DevOps sin un alineamiento efectivo entre los servicios que provee la organización a los usuarios y clientes con los servicios de los componentes que intervienen en el desarrollo y despliegue de aplicaciones. Por lo tanto es necesario describir Metodologías y Modelos de alineamiento que ayuden a efectivizar la los objetivos empresariales a mediante de TI.
- Se describió el desarrollo y despliegue de aplicaciones basados en la regulación de las actividades a partir del análisis de los conceptos específicos de los principios de DevOps con su estructura apoyados en integración, entrega y despliegue continuo, así como las fronteras de los roles basados en compartición, automatización y métricas, que logren regular actividades sobre procesos sensibles, permitiendo agilidad en la entrega de prestaciones al usuario.
- Se ha propuesto un modelo de incorporación de herramientas al ciclo de vida con DevOps. en base a criterios de disponibilidad, control de gestión, integración continua y versionamiento de las aplicaciones.
- Se ha propuesto un Protocolo para control de software en producción. Las herramientas DevOps y un control ordenado de monitoreo logran aporte de disponibilidad de servicios a los usuarios.
De esta manera se logra alineamiento efectivo entre la Organización y TI mediante el uso adecuado de los recursos de TI.
- Se ha planteado un modelo de buenas prácticas apoyado en ITIL, en la Metodología *Lean Startup* y las Métricas, a fin de alinear las fases del ciclo de

vida en base a DevOps, que permita efectividad en cada etapa del ciclo de vida del desarrollo y despliegue de aplicaciones.

- Se ha propuesto un modelo para la administración eficiente de recursos de un Data Center en forma gráficas y en base a una hoja de ruta, que sirva de guía para la implantación del modelo SAM-RS y de la implementación de las herramientas DevOps.
- Se ha planteado el modelo de aplicación del MAS-RS y el modelo incorporación de las herramientas DevOps en los componentes del Data Center, en dos casos prácticos. Se presentan resultados para cada caso.
La aplicación de los conceptos fundamentales de las metodologías SAM-RS y Lean Startup, el Ciclo de Vida DevOps, Herramientas DevOps con fundamentos metodológicos ayudan a verificar que el resultado es beneficioso para el Gobierno Corporativo, TI, usuarios internos y clientes externos.

7.2. Trabajos Futuros

- I. Evaluación de las Métricas DevOps que permitan una caracterización con valor agregado en cada fase del Ciclo de Vida DevOps, que ayuden al alineamiento entre la organización y TI.
- II. Integración del Servicio de Seguridades de ITIL en los componentes del Data Center mediante Herramientas DevOps.
El Servicio de Gestión de Seguridades de ITIL ayuda a establecer la integridad y confidencialidad de los datos e información.
La correlación de la Gestión de Seguridades en cada etapa del Ciclo de Vida DevOps, apoyado por el Modelo de Alineamiento Estratégico y la Metodología Lean Startup, permitirá estructurar un modelo de desarrollo y despliegue de software seguro y confiable con normas que garanticen disponibilidad de la información a los usuarios.
- III. Determinar el acompañamiento de otros Marcos de Referencia o buenas prácticas como Normas ISO 38500, para evaluar mejores alineamientos entre la Organización y TI, en el marco de tareas colaborativas con DevOps.

8. BIBLIOGRAFÍA

- Akerele, O., Ramachandran, M., & Dixon, M. (2014). Evaluating the Impact of Critical Factors in Agile Continuous Delivery Process: A System Dynamics Approach. *International Journal of Advanced Computer Science and Applications*. Last visit: March 30, 2017, retrieved from <http://doi.org/10.14569/IJACSA.2014.050319>
- Álvarez, M. (2015). Qué es Vagrant, cómo trabajar con Vagrant. Last visit: March 30, 2017, retrieved from <http://www.desarrolloweb.com/articulos/trabajar-con-vagrant.html>
- Avison, D., Jones, J., Powell, P., & Wilson, D. (2004). Using and validating the strategic alignment model. *Journal of Strategic Information Systems*, 13(3), 223–246. Last visit: April 30, 2017, retrieved from <http://doi.org/10.1016/j.jsis.2004.08.002>
- Belalcázar, A. (2015). Control y Recuperación de componentes de Data Center UNLP. ISBN: 978-987-1896-47-9, Last visit: April 30, 2017, retrieved from <http://conaiisi2015.utn.edu.ar/memorias.html>. 1 - 8.
- Belalcázar, A., Díaz, J., & Molinari, L. (2015). Towards the Strategic Alignment of Corporate Services with IT, applying Strategic Alignment Model (SAM), *Journal of Computer Science and Technology (JCS&T)*, ISSN: 1666-6038 (Online), 1666-6046 (Print), Last visit: April 27, 2017, retrieved from <http://journal.info.unlp.edu.ar/wp-content/uploads/2015/10/JCST-42-Paper-7.pdf>. 1 - 7.
- Belalcázar, A., Díaz, J., & Molinari, L. (2016). Principios, Roles y Métricas en alineamiento estratégico de nuevos requerimientos utilizando DevOps. Congreso Internacional CACIC 2016. ISBN: 978-987-733-072-4, Universidad Nacional San Luis Argentina. Libro de Actas. 1042 - 1051.
- Blank, S., & Dorf, B. (2012). *The Startup Owner's Manual*. Last visit: April 27, 2017, retrieved from <http://ctinnovations.com/images/resources/Startup%20Owners%20Manual%20-%20BlankDorf.pdf>
- Boar, B. (2001). *The Art of Strategic Planning for Information Technology*. Editorial: John Wiley & Sons, Inc. Second Edition
- Bourne, V. (2013). Measuring Your DevOps Success. Last visit: April 27, 2017, retrieved from <http://blog.appdynamics.com/devops/quantified-devops/>
- Brand, N., Beens, B., Vuuregge, E., & Batenburg, R. (2011). Engineering governance: introducing a governance meta framework. *Journal of Corporate Governance*: 2011, 2(2), 106-118. Last visit: April 15, 2017, retrieved from <http://doi.org/10.1504/IJCG.2011.041150>
- Canay, J. (2014). Gobierno de las Tecnologías de la Información en las Organizaciones. I Encuentro de Investigadores en Ciencias Económicas de Centro y Sur América Investigaciones Innovadoras para el Desarrollo Empresarial. Last visit April 18, 2017, retrieved from <http://www.www.gti4u.es/>
- Carzaniga, A., Fuggetta, A., Hall, R. S., Heimburger, D., van der Hoek, A., & Wolf, A. L. (1998). A Characterization Framework for Software Deployment Technologies. Technical Report. University of Colorado Department of Computer Science Technical Report CU-CS-857-98 April 1998 ?. Last visit: April 15, 2017, retrieved from http://scholar.colorado.edu/csci_techreports/806/
- Céspedes, J. (2015). Qué es el Gobierno de las TI. Universidad de Almería. Last visit: February 15, 2017, retrieved from <http://tic.crue.org/wp-content/uploads/2016/07/capitulo3.pdf>
- Chan, Y. (1992). Business Strategy, Information Systems Strategy, And Strategic Fit: Measurement And Performance Impacts. Last visit: April 20, 2017, <http://ir.lib.uwo.ca/cgi/viewcontent.cgi?article=3142&context=digitizedtheses>
- Chef. (2016). Achieve speed, scale, and consistency by automating your infrastructure with Chef. Last visit: November 30, 2016, retrieved from <https://www.chef.io/chef/>
- Cockburn, A., & Highsmith, J. (2001). Agile Software Development: The People Factor. *IEEE Computer*, 34(11), 131–133. Last visit: April 27, 2017, retrieved from <http://doi.org/10.1109/2.963450>
- Crom, J. (2005). *Curso Gestión de la Innovación*. Buenos Aires - Argentina. Last visit: April 15,

- 2017, retrieved from <http://www.ieee.org.ar/downloads/crom-2005-gestion.pdf>
- Davis, J., & Daniels, K. (2015). *Effective DevOps, Building a Culture of Collaboration, Affinity, and Tooling at Scale*. Editorial O'Reilly. Last visit: January 30, 2017. retrieved from <http://pdf.th7.cn/down/files/1602/Effective%20DevOps.pdf>
- De la Puente, A., & Zamorano, J. (2002). ORK: Tecnología de Software fiable para Sistemas de Tiempo Real,, 880(Rtc 1992). Universidad Politécnica de Madrid. Last visit: November 05, 2016, retrieved from <http://www.dit.upm.es/~str/papers/pdf/puente&06a.pdf>
- Fernández, A., & Llorens, F. (2009). An IT Governance Framework for Universities in Spain. EUNIS 2009 Conference, 1–13. Last visit: January 30, 2017, retrieved from http://gti4u.es/pdf/an_it_governance_framework_for_universities_in_spain.pdf
- Fernández, G. (2007). Seguridad en Sistemas Operativos. Universidad de la Coruña, Facultad de Informática. Last visit: February 25, 2017, retrieved from <http://sabia.tic.udc.es/docencia/ssi/old/2006-2007/docs/trabajos/03%20-%20Seguridad%20en%20Sistemas%20Operativos.pdf>
- Fernández, J., Mayol, E., & Pastor, J. A. (2008). Agile Business Intelligence Governance: Su justificación y presentación. Universitat Politècnica de Catalunya, (1), 1–7. Last visit: February 20, 2017, retrieved from http://itsmf.es/index.php?option=com_docman&task=doc_download&gid=764&Itemid=177
- Ferreira, J., Noble, J., & Biddle, R. (2007). Up-front interaction design in agile development. Proceedings of the 8th International Conference on Agile Processes in Software Engineering and Extreme Programming, 9–16. Last visit: April 20, 2017, retrieved from <http://doi.org/10.1007/978-3-540-73101-6>
- Flores, A. L. (2013). Aplicación del método Toyota a las iniciativas emprendedoras. Revista Digital de Acta. Last visit: April 20, 2017, retrieved from https://www.acta.es/medios/articulos/comercio_y_economia/011001.pdf
- Gaffney, S., Lin, S., Miller, K., Nilsson, H., Ravala, S., & Unnikrishnan, M. (2014). Lean Startup Methodology for Enterprises. How Established Companies Can Leverage Lean Startup Methodology for Sustaining and Disruptive Innovation. Engineering Leadership Professional Program (ELPP) - UC Berkeley, 1–16. Last visit: April 12, 2017, retrieved from <https://ikhlaqsidhu.files.wordpress.com/2014/08/elpp-project-1-lean-startup-for-enterprises-final-3.pdf>
- Gary, H., & Jimmy, H. (2008). Alineando Cobit 4.1 ITIL v3 e ISO/IEC 27002 en beneficio de la empresa. Governance An International Journal Of Policy And Administration, 1–130. Last visit: April 10, 2017, retrieved from http://www.isaca.org/Knowledge-Center/Research/Documents/Aligning-COBIT-ITIL-V3-ISO27002-for-Business-Benefit_res_Eng_1108.pdf
- Gfader, P. (2012). Use Scrum and Continuous Delivery to build the right thing. Last visit: February 20, 2017, retrieved from <https://www.scrum.org/resources/use-scrum-continuous-delivery-build-right-thing>
- Gimson, L. (2012). Metodologías ágiles y desarrollo basado en conocimiento. Trabajo final integrador para obtener el grado de Especialista en Ingeniería de Software, UNLP. Last visit: April 10, 2017, retrieved from http://sedici.unlp.edu.ar/bitstream/handle/10915/24942/Documento_completo_.pdf?sequence=1
- González, C. (2008). Generacion, Gestion Y Distribucion De Artefactos Java Con Tecnicas De Integracion Continua Y Software Libre, 85. Last visit: March 15, 2017, retrieved from http://oa.upm.es/1150/1/PFC_CARLOS_GONZALEZ_SANCHEZ.pdf
- Greer, D., & Ruhe, G. (2004). Software Release Planning: An Evolutionary and Iterative Approach. Information and Software Technology, 46(4)(4), 243-253. DOI: 10.1016/j.infsof.2003.07.002. Last visit: January 20, 2017, retrieved from <http://doi.org/10.1016/j.infsof.2003.07.002>
- Guske, M., & Kloss, M. (2009). Release a Day, Mythos oder Realität. Last visit: March 15, 2017, retrieved from <http://docplayer.org/1524009-Sonntag-4-oktober-2009-release-a-day->

- mythos-oder-realitaet.html
- Haight, C., & Bandopadhyay, T. (2015). DevOps basado en datos : El uso de la métrica como guía, 1–5. Last visit: April 22, 2017, retrieved from http://www.idglat.com/afiliacion/whitepapers/devops_basado.pdf?tk=/:
- Hamaker, B. S., & Hutton, A. (2004). Principles of IT Governance, 2–5.
- Henderson, J., & Venkatraman, N. (1990). Strategic Alignment: A Model for Organizational Transformation Via Information Technology. Center for Information Systems Research, Massachusetts Institute of Technology, 1–48.
- Hernantes, J., Gallardo, G., & Serrano, N. (2015). IT Infrastructure- Monitoring Tools. Software Technology. Last visit: December 25, 2016, retrieved from <http://docplayer.net/4019549-It-infrastructure-monitoring-tools.html>
- Hrabinski, S. (2009). Los SLAs y el uso de ITIL® en un contexto de outsourcing, Foro Global Crossing de Tecnología y Negocios. Last visit January 24, 2017, retrieved from <http://es.slideshare.net/foroglobalcrossing/hrabinski-iiforogrosario29abr09>
- Humble, J., & Farley, D. (2011). Continuous Delivery. Editorial Addison Wesley. Last visit: January 05, 2017, retrieved from <https://buildrelease.googlecode.com/hg-history/1998cd1d530b35b79740d7bf93f8915548136c25/Trunk/BreBooks/Continuous%2520Delivery.pdf>.
- Hüttermann, M. (2012). DevOps for Developers. Integrate Development and Operations, the Agile Way, Editorial Apress
- IEEE. (1990). IEEE Standard Glossary of Software Engineering Terminology. Last visit: April 20, 2017, retrieved from <http://ieeexplore.ieee.org/document/7435207/>
- IFAC. (2004). Enterprise Governance: Getting the Balance Right. International Federation of Accountants. Last visit: October 10, 2016, retrieved from http://www.cimaglobal.com/Documents/Research%20and%20Insight/2010-06-24-tech_execrep_enterprise_governance_getting_the_balance_right_feb_2004.pdf
- Ingall, S. (2003). Working together fot Mental Health: A Proposed Mental Health Services Strategy for Newfoundland and Labrador, 44(0). Last visit: November 26, 2016, retrieved from http://www.health.gov.nl.ca/health/publications/mental_health_strategy_disc_doc_nov_2003.pdf
- ISACA. (2017). COBIT 5. Last visit: April 15, 2017, retrieved from <https://www.isaca.org/COBIT/Pages/COBIT-5-spanish.aspx>
- ITIL Official. (2015). Official Site. Last visit: April 23, 2017, retrieved from <http://www.itil-officialsite.com>
- Jiménez, M. (2011). Alineamiento de Estratégico de TI con el Negocio. San José Costa Rica. Last visit: January 15, 2017, retrieved from http://www.clubinvestigacioncr.com/docs/042_alineamiento-de-estrategico-de-ti-con-el-negocio.pdf
- Kaplan, R., & Norton, D. (2004). Mapas Estrategicos. Convirtiendo los Activos Intangibles en Resultados Tangibles. Un libro de Harvard Business School Press.
- Lara, M. (2013). Pruebas de software, 1–34. Last visit: April 20, 2017, retrieved from https://netbeans.org/index_es.html
- Lent, J. (2014). DevOps Definition: Best explained by what it’s not. Last visit: April 25, 2017, retrieved from http://docs.media.bitpipe.com/io_11x/io_110410/item_746948/CA_sSoftwareQ_IO%23110410_Eguide_080713_LI%23746948.pdf
- López, D. (2012). DataCenter Diseño Sostenible. BICSI Andino. 7 Congreso y Muestra Comercial. Last visit: April 20, 2017, retrieved from https://www.bicsi.org/uploadedFiles/BICSI_Website/Global_Community/Presentations/An-dean/DIA%201%20CONF%202%20HUBBELL.pdf

- Lopez Paz, C., Macia Péres, F., & Delgado Fernández, M. (2008). El problema de alinear las Tecnologías de la Información con el Negocio. Last visit: April 20, 2017, retrieved from ccia.cujae.edu.cu/index.php/siia/siia2008/paper/download/1204/268
- Luque, E., & Rexachs, D. (2014). High Performance Computing for Efficient Applications and Simulation. Universidad Autónoma de Barcelona. Seminario Cómputo de Altas Prestaciones, brindado en la Facultad de Informática de la Universidad Nacional de la Plata Argentina Agosto 2014.
- Luthiger, B. (2015). DevOps bei den ID Build-Automatisierung statt Silo-Betrieb Ausgangslage, 1–14. Last visit: April 20, 2017, retrieved from <http://docplayer.org/4580780-Devops-bei-den-id-build-automatisierung-statt-silo-betrieb.html>
- Marulanda Echeverry, C. E., López Trujillo, M., & Cuesta Iglesias, C. A. (2009). Modelos de Desarrollo para Gobierno TI. Models of development for IT government. *Scientia et Technica*, 1(41), 185–190. Last visit: March 30, 2017, retrieved from <https://dialnet.unirioja.es/servlet/articulo?codigo=4728957>
- Medina, Y., & Rico, D. (2009). Modelo de gestión basado en el ciclo de vida del servicio de la Biblioteca de Infraestructura de Tecnologías de Información (ITIL). *Revista Virtual Universidad Católica Del Norte*, (27), 1–21. Last visit: March 30, 2017, retrieved from <http://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/viewFile/105/208>
- Megias, J. (2013). Las 10 Métricas claves de una Startup, Estrategia, Startups y Modelos de Negocio. Last visit: March 30, 2017, retrieved from <http://javiermegias.com/blog/2013/05/metricas-startup-indicadores/>
- Molina, A., & Rosales, N. (2010). Identificación de competencias distintivas en la mejora de procesos., 2010. Last visit: August 27, 2016, retrieved from <http://monografias.umcc.cu/monos/2010/QUIMEC/mo1016.pdf>
- Montaña, A. (2013). Propuesta para la implementación de un esquema de gobierno de TI en ambientes tercerizados, 216. Last visit: February 18, 2017, retrieved from <http://www.bdigital.unal.edu.co/11568/1/2822122.2013.pdf>
- Muñoz, L., & Ulloa, G. (2011). Gobierno de TI—Estado del arte. *Sistemas & Telemática*, 9, 23–53. Last visit: February 20, 2017, retrieved from http://www.icesi.edu.co/revistas/index.php/sistemas_telematica/article/view/1052
- Myers, G. (2008). *The Art of Software Testing*. Editorial: John Wiley & Sons, Inc. Second Edition.
- OECD. (2015). *Principles of Corporate Governance*, (Principles of Corporate Governance). Last visit: January 18, 2017, retrieved from <https://www.oecd.org/daf/ca/Corporate-Governance-Principles-ENG.pdf>
- Ojo, A., Janowski, T., Shareef, M., & Estevez, E. (2009). Aligning Electronic Government and Public Administration Reform Programs - Process, Tool and Case Study. UNU-IIST Report. Last visit April 20, 2017, retrieved from <https://www.researchgate.net/publication/264844706>
- Ojo, A., Pratap B, Janowski, T. (2009). Strategic IT Planning for Public Organizations: A Toolkit. Last visit: April 20, 2017, retrieved from <http://i.unu.edu/media/unu.edu/publication/1370/report417.pdf>
- Ordiz, M., & Pérez, G. (2002). Creación de valor en la empresa a través de las tecnologías de la información y comunicación. Universidad de Oviedo. 2002-2011, 1–14. Last visit: November 10, 2016, retrieved from http://www.esic.edu/documentos/revistas/esicmk/060130_574899_E.pdf
- Pardo, C., Hurtado, J. A., & Collazos, C. A. (2010). Mejora de Procesos de software ágil con Agile-SPI Process. *Revista DYNA*, 77, 251–263. Last visit: November 20, 2016, retrieved from http://www.researchgate.net/publication/265088413_Mejora_de_Procesos_de_software_gil_con_Agile-SPI_Process
- Paul M. Dooley. (2015). *The Intersection of DevOps and ITIL*. Expert Reference Series of White Papers. Global Knowledge. Last visit: April 20, 2017, retrieved from www.globalknowledge.net
- Pérez, D. (2007). Inversión en TIC y medición de sus beneficios. El caso de las Pyme del sector de las tecnologías de la información y las comunicaciones. *Boletín ICE Económico*, (2902), 65–

78. Last visit: February 20, 2017, retrieved from http://www.revistasice.info/cache/pdf/BICE_2902_65-80_E1C730CE2F4DE0D58CDDA8C19DC12CEE.pdf
- Pracht, U. (2011). Menschen und IT. DevOps - Gemeinsam Produktiv Werden. Last visit: February 20, 2017, retrieved from https://menschenundit.files.wordpress.com/2012/10/devops-artikel_udopracht.pdf
- Pracht, U. (2012). DevOps. Last visit: February 20, 2017, retrieved from http://www.sigs.de/download/oop_2012/files/Di2-2__Pracht_DevOps.pdf
- Pracht, U., & Architektur, J. (2011). Meine oder deine Architektur ? Von der Entwicklung zum Betrieb mit DevOps, 1–6. Last visit: February 20, 2017, retrieved from https://www.sigs-datacom.de/uploads/tx_dmjournals/pracht_OS_Architekturen_11.pdf
- Rakitin, S. (2001). Software Verification and Validation for Practitioners and Managers. Second Edition. Last visit: January 15, 2017, retrieved from WWW.BZUPAGES.COM
- Raxl, & Assaff, P. De. (2011). Hacia una Gestión Efectiva de Data Centers. Last visit: March 12, 2017, retrieved from http://www.cronista.com/documentos/Raxl_Assaff_-_ITBA.pdf
- Relic, N. (2014). Navigating Devops, What it is and why it matters to you and your business. Last visit: April 11, 2017, retrieved from https://try.newrelic.com/rs/412-MZS-894/images/NewRelic_DevOps-101-Navigating-DevOps-eBook.pdf
- Ries, E. (2011). The Lean Startup. Last visit: January 10, 2016, retrieved from <http://theleanstartup.com/>
- Ries, E. (2016). The Lean Startup Methodology. Last visit: January 12, 2016, retrieved from <http://theleanstartup.com/principles>
- Ríos Huércano, S. (2014). Manual De Itil V3, V3, 1–101. Last visit: April 20, 2017, retrieved from <http://www.biabile.es/wp-content/uploads/2014/ManualITIL.pdf>
- Ritegno, E. O. (2014). Gestión de Riesgos de TI Un enfoque desde el marco de trabajo, 1–17. Last visit: April 18, 2017, retrieved from <http://docplayer.es/9130060-Gestion-de-riesgos-de-ti-un-enfoque-desde-el-marco-de-trabajo-cobit-5-eduardo-oscar-ritegno-banco-de-la-nacion-argentina.html>
- Rivadeneira, S. (2012). Metodologías ágiles enfocadas al modelado de requerimientos. Universidad Nacional de La Patagonia Austral, 5(1). Last visit: February 25, 2017, retrieved from <https://dialnet.unirioja.es/descarga/articulo/5123612.pdf>
- Robinson, A. (2016). InfoSec Reading Room Continuous Security : Implementing the Critical Controls in a DevOps Environmen. Last visit: February 25, 2017, retrieved from <https://www.sans.org/reading-room/whitepapers/critical/continuous-security-implementing-critical-controls-devops-environment-36552>
- Rodríguez, C., Molinari, L., & Díaz, F. J. (2013). The Hard Way to Virtual Machine Administration : towards DevOps A Bridge between Developers and IT Operators, 13(3), 118–122. Last visit: February 22, 2017, retrieved from <http://sedici.unlp.edu.ar/handle/10915/34502>
- Ross, J. W., & Weill, P. (2002). Six IT Decisions Your IT People Shouldn't Make. Harvard Business. Last visit: January 30, 2017, retrieved from https://onlinecampus.bu.edu/bbcswebdav/pid-4078478-dt-content-rid-13837609_1/courses/16sprgmetcs782_ol/media/Ros02.pdf
- Rümmeler, T., & Schlag, C. (2014). DevOps und Continuous Delivery: sich gemeinsam kontinuierlich verbessern, 1–5. Last visit: April 15, 2017, retrieved from <http://docplayer.org/1627914-Devops-und-continuous-delivery-sich-gemeinsam-kontinuierlich-verbessern.html>
- Schäfer, A., Reichenbach, M., & Fey, D. (2013). IAENG Transactions on Engineering Technologies. Lecture Notes in Electrical Engineering. Last visit: January 15, 2017, retrieved from <http://doi.org/10.1007/978-94-007-4786-9>
- Schulze, R., & Adan, C. (2014). Gegensätze Ziehen Sich an oder Wie Man Mit DEVOPS Erfolgreich Brücken Bauen Kann, 2–7. Last visit: April 20, 2017, retrieved from <https://udf.de/files/it-consulting-unternehmensberatung/06-presse/02-veroeffentlichungen/mit-devops-erfolgreich-bruecken-bauen.pdf>
- Stankovic, J. A. (1988). A Serious Problem for Next-Generation Systems. Computer. Last visit:

- April 8, 2017, retrieved from <http://doi.org/10.1109/2.7053>
- Tejera, D., Alonso, A., & Miguel, M. A. De. (2011). Serialización en sistemas distribuidos de tiempo real crítico. Universidad Politécnica de Madrid. Last visit: January 6, 2017, retrieved from <http://www.dit.upm.es/~str/papers/pdf/tejera&07a.pdf>
- Toomey, M. (2009). Waltzing with the Elephant: A comprehensive guide to directing and controlling information technology. Last visit: November 15, 2016, retrieved from www.infonomics.com.au
- Tosar, S. (2013). Gobernanza Corporativa de TI: desafío imprescindible. Taller regional para Parlamentos Latinoamericanos. Montevideo Uruguay. Last visit: November 18, 2016, retrieved from http://www.ictparliament.org/sites/default/files/uruguay_-_silvia_tosar_0.pdf
- Tovar, J. C. (2013). Pruebas de Integración. Arquitectura e Integración de Sistemas de Software. Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla. Last visit: April 20, 2017, retrieved from <https://www.lsi.us.es/docencia/get.php?id=6731>
- Ubuntu. (2015). Ubuntu Daemons. Last visit: November 18, 2016, retrieved from <http://doc.ubuntu-es.org/Daemon>
- Van, E., Rensen, T., Saboerali, R., & Teng, E. (2016). Vagrant: Development environments made easy. Last visit, November 27, 2017, retrieved from <https://delftswa.github.io/chapters/vagrant/>
- Vargas, A., Boza, A., & Cuenca, L. (2011). Lograr la alineación estratégica de negocio y las tecnologías de la información a través de Arquitecturas Empresariales: Revisión de la Literatura, 1061–1070. Last visit: February 8, 2017, retrieved from https://www.researchgate.net/publication/232715314_Lograr_la_alineacion_estrategica_de_negocio_y_las_tecnologias_de_la_informacion_a_traves_de_Arquitecturas_Empresariales_Revision_de_la_Literatura
- Weill, P., Ross, J. W., No, C. W. P., & No, S. W. P. (2004). Center for Information System of Management Massachusetts Institute of Technology. Massachusetts IT Governance on One Page Peter Weill Research Article: a completed research article drawing on one or more CISR research projects that presents management f. Group, 18. Last visit: January 15, 2017, retrieved from <http://doi.org/10.1007/b115738>
- Weill, P., Subramani, M., & Broadbent, M. (2002). Center for Information System. Massachusetts Institute of Technology State Street Corporation: Evolving IT Governance Peter Weill and Richard Woodham. Last visit: January 20, 2017, retrieved from <http://dspace.mit.edu/bitstream/id/1422/4235-02.pdf/>