

MARCO DE TRABAJO ADAPTADO PARA PROYECTOS DE DESARROLLO
DE SOFTWARE USANDO TOGAF 9.1

LINA MARÍA ORTIZ OSPINA

UNIVERSIDAD PONTIFICIA BOLIVARIANA

ESCUELA INGENIERÍAS

FACULTAD DE INGENIERÍA EN TECNOLOGÍAS DE INFORMACIÓN Y
COMUNICACIÓN

MAESTRÍA EN TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

MEDELLÍN

2018

MARCO DE TRABAJO ADAPTADO PARA PROYECTOS DE DESARROLLO
DE SOFTWARE USANDO TOGAF 9.1

LINA MARÍA ORTIZ OSPINA

Trabajo de grado para optar al título de Magister en Tecnologías de
Información y Comunicación

Asesor

GLORIA LILIANA VÉLEZ

PhD

UNIVERSIDAD PONTIFICIA BOLIVARIANA

ESCUELA INGENIERÍAS

FACULTAD DE INGENIERÍA EN TECNOLOGÍAS DE INFORMACIÓN Y
COMUNICACIÓN

MAESTRÍA EN TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

MEDELLÍN

2018

DECLARACIÓN ORIGINALIDAD

“Declaro que esta tesis (o trabajo de grado) no ha sido presentada para optar a un título, ya sea en igual forma o con variaciones, en esta o cualquier otra universidad”. Art. 82 Régimen Discente de Formación Avanzada, Universidad Pontificia Bolivariana.

FIRMA AUTOR (ES)

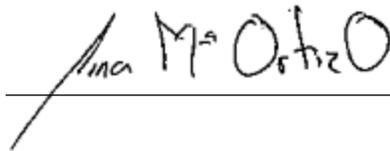
A handwritten signature in black ink, appearing to read 'M. Ortiz', is written over a horizontal line. The signature is slanted and includes a diagonal stroke on the left side.

Tabla de contenido

1. INTRODUCCIÓN	8
2. PLANTEAMIENTO DEL PROBLEMA	9
2.1. Problema	9
2.2. Justificación	10
3. OBJETIVOS.....	11
3.1. Objetivo General.	11
3.2. Objetivos Específicos	11
4. MARCO REFERENCIAL	12
4.1. Marco contextual	12
4.2. Marco Conceptual	12
4.2.1. Arquitectura empresarial.....	13
4.2.2. Lenguajes de modelado	18
4.2.3. Elicitación de requisitos	25
4.3. Estado del arte	27
4.3.1. Arquitectura empresarial.....	27
4.3.2. Técnicas de elicitación de requisitos.....	33
5. METODOLOGÍA.....	37
6. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS	38
6.1. Objetivo 1 - Documentación de apoyo de la metodología	38
6.1.1. TOGAF	39
6.1.2. Archimate.....	39
6.1.3. BPMN 2.0	76
6.1.4. UML 2.0	82
6.1.5. SCRUM	88
6.1.6. Elicitación de requisitos	93
6.2. Objetivo 2 – Marco de trabajo adaptado	99
6.2.1. Fase preliminar.....	101
6.2.2. Fase A - Visión de la arquitectura	105
6.2.3. Fase B - Arquitectura del negocio	111
6.2.4. Fase C - Arquitectura de sistemas de información	118
6.2.5. Fase D - Arquitectura tecnológica.....	135
6.2.6. Fase E - Oportunidades y soluciones	139
6.2.7. Fase F - Planificación de la migración	143
6.2.8. Fase G - Gobierno de la implementación.....	147
6.2.9. Fase H - Gestión del cambio de la arquitectura	154
6.2.10. Gestión de requerimientos	156
6.2.11. Arquitecturas de referencia	160

6.3. Objetivo 3- Implementación del marco de trabajo adaptado	167
6.3.1. Documentos generados	168
6.4. Objetivo 4 - Conclusiones del uso del marco de trabajo adaptado	168
7. CONCLUSIONES.....	169
7.1.1. Cómo apoya la arquitectura el marco de trabajo adaptado.....	169
7.1.2. Fortalezas del marco de trabajo adaptado	170
7.1.3. Oportunidades de mejora.....	171
8. REFERENCIAS.....	173
ANEXO 1.....	179

Tabla de ilustraciones

Ilustración 1. Análisis comparativo de los marcos de Arquitectura Empresarial	14
Ilustración 2. Ciclo de método de desarrollo de arquitectura	16
Ilustración 3. Relación entre Togaf 9.1 y Archimate 3.0	19
Ilustración 4. Relación entre BPM, Archimate, Ingeniería de software y Togaf	29
Ilustración 5. Alineación de arquitectura e ingeniería al inicio del ciclo de vida	30
Ilustración 6. Mapa de patrones para el artículo	31
Ilustración 7. Resultado de la evaluación cualitativa y las técnicas de elicitación de requisitos	35
Ilustración 8. Wiki Implementada con la base teórica de la metodología	38
Ilustración 9. Esquema de documentación del marco de trabajo	40
Ilustración 10. Relación entre Togaf 9.1, Archimate 3.0, UML 2.0, BPMN 2.0 y otras técnicas de elicitación de requisitos	41
Ilustración 11. Para este ejemplo el servicio que se presta son las compra por internet	42
Ilustración 12. Para acceder a un servicio de compras por internet es necesario contar con un PC, Smartphone o Tablet (Estas son las interfaces)	43
Ilustración 13 Funciones que debe realizar el sistema de información para realizar la venta	43
Ilustración 14. Ejemplo de proceso	44
Ilustración 15. Notación de actor de negocio	46
Ilustración 16. Notación de rol de negocio	46
Ilustración 17 Notación de la colaboración de negocio	47
Ilustración 18 Notación de la interface de negocio	47
Ilustración 19. Notación del proceso de negocio	48
Ilustración 20. Notación de la función de negocio	48
Ilustración 21. Notación de la interacción de negocio	49
Ilustración 22. Notación de evento de negocio	50
Ilustración 23. Notación de servicio de negocio	50
Ilustración 24. Notación de objetivo de negocio	51
Ilustración 25. Notación de contrato	51
Ilustración 26. Notación de representación	52
Ilustración 27. Notación de producto	52
Ilustración 28. Metamodelo de la capa de negocios	53
Ilustración 29. Notación de componente de aplicación	54
Ilustración 30. Notación de colaboración de aplicación	55
Ilustración 31. Notación de interface de aplicación	55
Ilustración 32. Notación de función de aplicación	56
Ilustración 33. Notación de interacción de aplicación	57
Ilustración 34. Notación de proceso de aplicación	57
Ilustración 35. Notación de evento de aplicación	58
Ilustración 36. Notación de servicio de aplicación	58
Ilustración 37. Notación de objeto de datos	59
Ilustración 38. Metamodelo de la capa de aplicaciones.....	60
Ilustración 39. Notación de nodo	61
Ilustración 40. Notación de dispositivo	61
Ilustración 41. Notación del software del sistema	62

Ilustración 42. Notación de colaboración tecnológica	63
Ilustración 43. Notación de interface tecnológica	63
Ilustración 44. Notación de Path	64
Ilustración 45. Notación de redes	64
Ilustración 46. Notación de función tecnológica	65
Ilustración 47. Notación de proceso tecnológica	65
Ilustración 48. Notación de interacción tecnológica	66
Ilustración 49. Notación de evento tecnológico	67
Ilustración 50. Notación de servicio tecnológico	67
Ilustración 51. Notación de artefacto tecnológico	68
Ilustración 52. Metamodelo de la capa de tecnología.....	69
Ilustración 53. Notación de interesado	70
Ilustración 54. Notación de conductor	70
Ilustración 55. Notación de evaluación	71
Ilustración 56. Notación de objetivo	71
Ilustración 57. Notación de restricción	71
Ilustración 58. Notación de principio	72
Ilustración 59. Notación de valor	72
Ilustración 60. Metamodelo de la capa de motivación	73
Ilustración 61. Notación de paquete de trabajo	74
Ilustración 62. Notación de entregable	74
Ilustración 63. Notación de meseta	75
Ilustración 64. Notación de brecha	75
Ilustración 65, Metamodelo de la capa de implementación y migración	76
Ilustración 66. Resumen de eventos en BPMN 2.0	78
Ilustración 67. Resumen de actividades	79
Ilustración 68. Resumen de compuertas	80
Ilustración 69. Resumen de artefactos	82
Ilustración 70. Marco de trabajo de Scrum	89
Ilustración 71. Formato de historia de usuario para el marco de trabajo adaptado.....	95
Ilustración 72. Ciclo del método de desarrollo de arquitectura	100
Ilustración 73. Fase preliminar.....	101
Ilustración 74. Evidencia de la declaración del trabajo de arquitectura	105
Ilustración 75. Fase A - Visión dela solución	106
Ilustración 76. Evidencia Visión de la arquitectura.....	110
Ilustración 77. Evidencia de principios.....	110
Ilustración 78. Evidencia de las actas de aprobación.....	110
Ilustración 79. Evidencia del diagrama de motivación	111
Ilustración 80. Fase B - Arquitectura del Negocio.....	112
Ilustración 81. Formato de las historias de usuario para el marco de trabajo adaptado.....	116
Ilustración 82. Evidencia de las historias de usuario	118
Ilustración 83. Fase C - Arquitectura de sistemas de Información	119
Ilustración 84. Navegación a la arquitectura actual	128
Ilustración 84. Navegación a la arquitectura destino	129
Ilustración 86. Evidencia de las brechas	129
Ilustración 87. Navegación al diagrama de componentes.....	130
Ilustración 88. Navegación diagrama de capas.....	131

Ilustración 89. Evidencia del diagrama de despliegue.....	132
Ilustración 90. Navegación al diagrama de dominio.....	133
Ilustración 91. Evidencia del diagrama de estados.....	134
Ilustración 92. Evidencia de prototipos	135
Ilustración 93. Fase D - Arquitectura de Tecnología	136
Ilustración 94. Fase E - Oportunidades y Soluciones	140
Ilustración 95. Evidencia de mapas de ruta	143
Ilustración 96. Fase F - Planeación de la migración	144
Ilustración 97. Evidencia del plan de implementación y migración	147
Ilustración 98. Fase G - Gobierno de implementación	148
Ilustración 99. Evidencia de las auditorías.....	153
Ilustración 100. Evidencia de planeación de sprint	153
Ilustración 101. Fase H - Gestión del Cambio de Arquitectura	154
Ilustración 102. Evidencia de los controles de cambio	156
Ilustración 103. Gestión de Requerimientos	157
Ilustración 104. Navegación a los requisitos funcionales y no funcionales.....	160
Ilustración 105. Arquitectura de aplicaciones web, diagrama de capas	161
Ilustración 106. Arquitectura de aplicaciones web, diagrama de despliegue	162
Ilustración 107. Arquitectura móvil, diagrama de capas	163
Ilustración 108. Arquitectura de aplicaciones móviles. Diagrama de distribución.....	164
Ilustración 109. Arquitectura de integración.....	165
Ilustración 110. Observaciones sobre la arquitectura de integración.....	165
Ilustración 111. Navegación a la Arquitectura de referencia de infraestructura	166
Ilustración 113. Navegación a la Arquitectura de referencia de seguridad.....	167

GLOSARIO

Archimate: The Open Group (2013) Describe Archimate como un estándar que provee un lenguaje gráfico para la representación de la arquitectura empresarial.

Arquitectura empresarial: ISO/IEC 42010:2001 define la arquitectura como *“La organización fundamental de un sistema, compuesta por sus componentes, las relaciones entre ellos y su entorno, así como los principios que gobiernan su diseño y evolución”*

BPMN: Lenguaje de modelado de procesos

QAW: Taller que ayuda a identificar los requisitos no funcionales que el arquitecto debe tener en cuenta para el diseño del Sistema de información empresarial

Requerimientos de software: Necesidades funcionales y no funcionales que debe implementar un sistema de información

Scrum: Metodología ágil que permite centrarse en lo importante para dar paso a un producto de alto valor.

Togaf: Marco de referencia de arquitectura empresarial, es una herramienta alinear la estrategia de la empresa con la tecnología de la misma.

UML: Grupo de diagramas que permiten documentar diversos modelos de *software*

RESUMEN

Este trabajo se soporta en la investigación del marco de trabajo Togaf 9.1 y los lenguajes de modelados Archimate 3.0, UML 2.0 y BPMN 2.0 con el fin de adaptar Togaf 9.1 a los proyectos de desarrollo de *software* y garantizar la alineación de estos con la estrategia de la compañía. Dicha alineación se realizará mediante talleres, herramienta, formatos y una estructura de documentación que agrupe los lenguajes de modelado mencionados anteriormente.

A partir de los entregables definidos se podrá diagramar el modelo motivacional, las arquitecturas de negocio, aplicaciones, datos e infraestructura y la relación entre ellas, adicionalmente se podrán visualizar brechas, capacidades arquitectónicas de la organización y gobierno arquitectónico.

PALABRAS CLAVE Arquitectura empresarial, desarrollo de *software*, ArchiMate, TOGAF

ABSTRACT

This work is supported by the research of the *framework* TOGAF 9.1 and the modeling languages ArchiMate 3.0, UML 2.0 and BPMN 2.0; to adapt TOGAF 9.1 to *software* development projects to ensure project alignment with companies' strategy. Such alignment will be done through workshops, tools, formats and structured documentation grouping the modeling languages mentioned above.

From the defined deliverables, could be diagramed the motivational model, business architectures, applications; data and infrastructure, the relationship between them, gaps, architectural capabilities of the organization and architectural government.

KEY WORDS: Enterprise architecture, software development, ArchiMate, TOGAF

1. INTRODUCCIÓN

El proyecto consiste en el estudio del marco de trabajo Togaf 9.1 y cómo este apoya los proyectos de desarrollo de *software*, permitiendo alinear el negocio con las tecnologías de información. Como resultado de esta investigación se entregará un marco de trabajo adaptado compuesto por talleres, herramienta, formatos y una estructura de documentación que agrupe los lenguajes de modelado ArchiMate 3.0, UML 2.0 y BPMN 2.0

2. PLANTEAMIENTO DEL PROBLEMA

2.1. Problema

Pereira, Serpa, & Rivas (2004) identifican siete factores de riesgo en un proyecto de desarrollo de *software* los cuales son: gestión, clientes y usuarios, requerimientos, estimación y programación de actividades, jefe de proyecto, proceso de desarrollo de *software* y personal de desarrollo.

Ayudando lo anterior y convencidos de que los requisitos son un factor clave en el desarrollo de la metodología se tienen dos afirmaciones que respaldan la hipótesis. La primera afirmación la hace Würfel, Lutz, & Diehl (2015) quien argumenta que los requisitos de *software* son especificaciones generadas por los usuarios y en muchas ocasiones no tienen conexión o la realidad, es decir, no están relacionados con la dimensión de negocio. Adicionalmente, los interesados pueden estar influenciados por soluciones del *software* existente desconociendo si estos realmente se adaptan a sus necesidades.

La segunda afirmación la realiza Pereira, Serpa, & Rivas (2004) quien afirma que entender los requerimientos no es una tarea sencilla y es un factor crítico en el desarrollo de los sistemas de información, ya que en muchas ocasiones los usuarios no logran entender los problemas a los cuales se están enfrentando. Adicionalmente sugiere que los requisitos deben ser identificados de forma temprana para disminuir los riesgos.

Desde la experiencia como profesional se ha identificado que los proyectos de desarrollo de *software* tienen un alto control de cambios, lo que genera sobre ejecución del presupuesto, reprocesos y retrasos para la salida a producción. Una solución a este problema es desarrollar los proyectos usando marcos de trabajo de arquitectura empresarial, para este proyecto se trabajará con Togaf 9.1. porque entrega buenas prácticas que buscan la alineación de la estrategia, los procesos de negocio, los sistemas de información, los datos y la tecnología de la empresa.

En el marco de trabajo de Togaf 9.1 los requisitos funcionales son evaluados en las dimensiones de negocio, sistema de información y datos, con el fin de garantizar que todos respondan a las necesidades del negocio y la estrategia de la empresa. Adicionalmente los requisitos no funcionales son el insumo fundamental para desarrollar los diseños arquitectónicos de los sistemas de información y de esta forma garantizan el cumplimiento de los atributos de calidad, los cuales se encuentran identificados y alineados en las capas de sistemas de información y tecnología.

2.2. Justificación

Teniendo en cuenta el problema planteado se pretende desarrollar un marco de trabajo adaptado con sus respectivas herramientas, que apoye al desarrollo de *software* empresarial basado en el *framework* de arquitectura empresarial Togaf 9.1; garantizando que los requisitos identificados y el diseño arquitectónico estén orientado y alineado con la estrategia de la empresa y las dimensiones de negocio, aplicaciones, datos e infraestructura.

La entrega de la documentación o generación de artefactos se enfocará principalmente en lenguajes de modelado, que reflejen la trazabilidad y relación de todos los componentes involucrados. Ernst (2008) resalta que documentar al gestionar la arquitectura empresarial es un tema avanzado, ya que a menudo incluye una gran cantidad de aplicaciones y soluciones de negocios, incluyendo la integración entre ellas. De este modo, la gestión de la arquitectura empresarial es una tarea que debe ser ejecutada en empresas que tengan la tecnología como soluciones y herramientas verticales a sus procesos.

Napoli & Kaloyanova (2011) afirma que durante la última década la tendencia es hacia un enfoque centrado en el negocio. Los procesos de desarrollo de *software* se basan en modelos de negocio más flexibles promoviendo la integración entre sistemas de información, generando así la evolución en las tecnologías de información, tales como arquitectura orientada a servicios, arquitectura empresarial, Procesos de Negocio y Gestión de Servicios.

Por la afirmación anterior, la metodología busca alinear la estrategia de la empresa y los procesos afectados con el desarrollo de *software* empresarial, para las dimensiones de datos, aplicaciones e infraestructura. Adicional a lo anterior los interesados del proyecto tendrán una visión más amplia de los factores externos que afectan el proyecto, procesos con los que interactúa, aplicaciones con las cuales el nuevo desarrollo debe integrarse, requisitos de calidad importantes para el negocio y requisitos funcionales del negocio.

Adicionalmente, el marco de trabajo adaptado tendrá como referencia el ADM, repositorio de arquitectura, marco de contenido, capacidades y gobierno de Togaf 9.1 y la documentación se realizará sobre lenguajes de modelado UML 2.0, BPMN 2.0 y Archimate 3.0. En cada una de las fases del ADM de Togaf se tendrán entregables específicos orientados al producto de *software*, donde se identifiquen los factores motivacionales, la arquitectura base, la arquitectura objetivo y las arquitecturas de transición. El repositorio de arquitectura tendrá un conjunto de arquitecturas de referencia de seguridad, integración, aplicaciones, infraestructura entre otros.

3. OBJETIVOS

3.1. Objetivo General.

Desarrollar una metodología que permita la alineación en el proceso de construcción de *software* empresarial con el modelo de negocio de la empresa empleando Archimate 3.0, BPMN 2.0 y UML 2.0 como lenguaje de modelado y alineados con el marco de trabajo Togaf 9.1

3.2. Objetivos Específicos

- Analizar las herramientas y técnicas de Togaf 9.1 que pueden ser aplicadas al desarrollo de *software* empresarial
- Diseñar la metodología que permita la identificación del alcance de los proyectos de desarrollo de *software* alineados con Togaf 9.1
- Implementar la metodología planteada en un caso de estudio para la validación de los resultados y la aceptación por parte del equipo de trabajo
- Evaluar la metodología planteada mediante los resultados del caso de estudio, que permita la validación del éxito de la misma o la identificación de brechas

4. MARCO REFERENCIAL

4.1. Marco contextual

La tesis de grado se aplicará en proyectos de desarrollo de aplicaciones empresariales. El foco principal de aplicación son empresas de desarrollo de *software* o pequeñas fábricas de *software* al interior de grandes empresas, con el fin de aumentar el valor en sus entregas y garantizar que todos los módulos o soluciones estén enfocados en la estrategia de la compañía, de esta forma las soluciones serán aprobadas desde la dirección, por lo que tendrán más visibilidad y apoyo.

4.2. Marco Conceptual

El marco conceptual agrupa los temas que soportan el trabajo de investigación que se está proponiendo. Togaf 9.1 es el marco de trabajo de arquitectura empresarial seleccionado, este marco ofrece buenas prácticas para alinear la estrategia de la compañía con las tecnologías de la información, conociendo el estado actual y el estado deseado en las dimensiones de procesos, datos, aplicaciones y/o tecnología, adicionalmente entrega un mapa de ruta que contiene los proyectos para alcanzar el estado deseado. Esta propuesta se centrará en los proyectos de desarrollo de *software* empresarial que están enmarcados en el mapa de ruta, con el fin de garantizar que su desarrollo esté en el marco estratégico de la compañía y garantizar que los datos y la infraestructura modelada satisfagan las necesidades de los procesos y la estrategia.

Para identificar los requisitos funcionales y atributos de calidad del *software* a desarrollar es necesario conocer cuáles son las técnicas más usadas y con mejores resultados, de esta forma se garantiza que los requisitos identificados están alineados con los procesos de la compañía y no con el deseo del usuario final. Adicionalmente, este es el insumo principal para realizar el diseño detallado y el diseño arquitectónico del sistema de información empresarial.

Por último, toda la información identificada en el proceso de arquitectura empresarial, la elicitación de requisitos, el diseño detallado y el diseño arquitectónico deben estar modelados en lenguajes que permitan la trazabilidad y el complemento entre ellos mismos, para lo que se propone Archimate 3.0, BPMN 2.0 y UML 2.0

4.2.1. Arquitectura empresarial

ISO/IEC 42010:2001 define la arquitectura como “*La organización fundamental de un sistema, compuesta por sus componentes, las relaciones entre ellos y su entorno, así como los principios que gobiernan su diseño y evolución*” basados en esta definición se puede afirmar que la arquitectura empresarial es una visión general y global de una compañía, donde se identifican cada uno de sus componentes y cómo se relacionan para dar valor a la empresa.

Chiprianov, Kermarrec, & Rouvrais (2012) expresa que la arquitectura empresarial es beneficiosa para entender la complejidad del sistema, alinear el negocio con las plataformas tecnológicas, la interoperabilidad y la integración de los sistemas que constituyen y genera un entendimiento común en toda la empresa.

Los componentes al interior de una compañía están compuestos por su estrategia, cadena de valor, procesos, datos, sistemas de información e infraestructura. Cuando estos componentes interactúan entre sí y se encuentran alineados, se aumenta la posibilidad de alcanzar los objetivos estratégicos.

Benavides & Galindo (2013) y Chiprianov, Kermarrec, & Rouvrais (2012) resaltan que las teorías de arquitectura empresarial se encuentran diversos *framework* como son Zachman, Marco federal de arquitectura empresarial, método Gartner, Togaf, entre otros, donde cada creador identifica diferentes capas, herramientas y metodologías para ser aplicado.

Basado en el estudio realizado por Sessions (2007) y Cabrera & Lopez (2014) se determina que Togaf 9.1 será el *framework* de referencia para trabajar en este proyecto. En su trabajo hicieron una comparación entre los siguientes marcos de trabajo: Zachman, Togaf, DODAF, Tafim, Feaf y Teaf, evaluando las siguientes características: Integridad de la taxonomía, integridad del proceso, modelo de referencia, guía práctica, madurez del modelo, orientación del negocio, orientación del gobierno, catálogo prescriptivo, neutralidad del proveedor, disponibilidad de la información y una valoración del modelo en una escala del 1 al 5, siendo 5 la mejor valoración. La comparación arrojó el siguiente resultado:

Análisis comparativo de los Marcos de Arquitectura Empresarial						
Criterio	Zachman	DODAF	TAFIM	FEAF	TEAF	TOGAF
Integridad de la Taxonomía	4	2	2	2	1	2
Integridad del proceso	1	3	2	2	3	4
Modelo de referencia	1	2	2	4	1	3
Guía práctica	1	4	3	2	4	4
Madurez del modelo	1	2	3	3	2	1
Orientación de negocio	1	4	3	1	4	2
Orientación de gobierno	1	3	4	3	3	2
Catálogo prescriptive	1	2	3	4	2	2
Neutralidad del proveedor	2	1	1	3	1	4
Disponibilidad de la información	2	1	1	2	1	4
Valoración del modelo	1	2	2	1	2	3
Totales	16	26	26	27	24	31

Ilustración 1. Análisis comparativo de los marcos de Arquitectura Empresarial ¹

4.2.1.1. *Togaf*

El Open Group define Togaf como un marco de referencia de arquitectura, es una herramienta para guiar la aceptación, creación, uso y mantenimiento de arquitecturas. Es un modelo iterativo apoyado en las mejores prácticas con un conjunto reutilizable de activos arquitectónicos existentes.

Este marco fue creado en 1995 y desde ese momento ha venido evolucionando, actualmente se encuentra en la versión 9.1 la cual fue publicada en diciembre de 2001. Esta es la versión de referencia en la cual el proyecto se va a desarrollar.

Togaf cubre cuatro tipos de arquitectura, permitiendo de esta forma alinear el negocio con las tecnologías de información, y permitiendo la comunicación de

¹ (Cabrera & Lopez, 2014)

todos sus componentes. Los tipos de arquitectura que maneja Togaf son los siguientes:

- Arquitectura de negocio. “*La estrategia de negocio, gobierno, organización y procesos clave de la organización*”. (Josey, 2013)
- Arquitectura de datos. “*la estructura de datos lógicos y físicos que posee una organización y sus recursos de gestión de datos*”. (Josey, 2013)
- Arquitectura de aplicaciones. “*Un plano de las aplicaciones individuales a implementar, sus interacciones y sus relaciones con los procesos de negocio principales de la organización*”. (Josey, 2013)
- Arquitectura de tecnología. “*Las capacidades de software y hardware que se requieren para apoyar la implementación de servicios de negocio, dato y aplicación. Esto incluye infraestructura de IT, capa de mediación, redes, comunicaciones, procesamiento y estándares*”. (Josey, 2013)

Tupper (2011) Explica en su libro *Data Archimate* la importancia del *Continuum empresarial*, ya que permite simplificar el proceso de creación de arquitecturas. TOGAF ADM proporciona un procedimiento para la conducción de este movimiento desde el genérico a lo específico.

Las arquitecturas más genéricas se conocen como arquitectura Fundacional, esta puede ser utilizada las organizaciones de TI. El segundo nivel se conoce como arquitecturas comunes de sistemas, basadas en dominios específicos de arquitectura. El tercer nivel son las arquitecturas industriales, enfocados en dominios de la industria como la salud, farmacéutica, entre otros. Y finalmente el cuarto nivel es la arquitectura organizacional, está enfocada en una empresa en particular como resultado a sus necesidades específicas.

4.2.1.2. ADM Togaf

Según Tupper (2011) en su libro *Data Architect* el ADM o método de desarrollo de la arquitectura es el corazón del marco de trabajo, ya que a partir de este método las arquitecturas responden a los requerimientos del negocio.

Josey (2013) Coincide con Tupper (2011) en que el ADM es el núcleo de Togaf porque esta metodología describe un modo seguro para desarrollar una arquitectura empresarial, adicionalmente ofrece un conjunto de guías y técnicas para desarrollar diferentes arquitecturas como son la de negocios,

Teniendo en cuenta a Josey (2013) se explica de forma resumida el objetivo de cada una de las fases del ADM y que se busca cumplir en cada una de ellas:

- La fase preliminar prepara los procesos afectados por el proyecto de arquitectura empresarial para que entiendan y adopten la metodología. Adicionalmente se debe identificar el equipo de arquitectura, prepara la capacidad arquitectónica y seleccionar las herramientas que se van a utilizar en el proyecto.
- La fase de gestión de requerimiento es la encargada de identificar, almacenar, gestionar y priorizar los requisitos, interactúa todas las fases, y en caso de que se presente un cambio debe comunicar e identificar que otros requisitos fueron afectados por el cambio.
- La fase de visión de la arquitectura identifica los interesados, limitaciones y expectativas. Proporciona una visión general del alcance y los cambios que se realizarán en la compañía con el proyecto de arquitectura empresarial.
- En las fases de arquitectura de negocio, arquitectura de sistemas de información y arquitectura de tecnología se identifica la línea base de arquitectura o *AS-IS*, la arquitectura objetivo o *TO-BE* y la arquitectura de transición.
- En la fase de oportunidades y soluciones se realiza una planificación inicial para iniciar la ejecución de la arquitectura objetivo (*TO-BE*). Se identifican las arquitecturas de transición cuando hace un enfoque incremental del proyecto.
- En la fase de planificación de la migración se desarrolla el plan detallado para llegar a la arquitectura objetivo a partir de la línea base.
- En la fase de gobierno de la implementación se asegura que el proyecto de implementación se esté desarrollando de acuerdo al plan detallado. Adicionalmente se supervisa el alcance y se revisan las prioridades de implementación.
- En la fase de gestión de cambios de arquitectura se asegura que los cambios identificados en la ejecución del proyecto tengan un correcto procedimiento de aprobación luego de revisar los riesgos y cambios que genera en el resto del proyecto.

4.2.2. Lenguajes de modelado

4.2.2.1. Archimate

The Open Group (2013) Describe Archimate como un estándar que provee un lenguaje gráfico para la representación de la arquitectura empresarial. El lenguaje evoluciona de la mano de Togaf, es decir, cada vez que Togaf cambia de versión, Archimate crea nuevos componentes que permiten la diagramación de los nuevos conceptos.

The Open Group (2013) Explica que el *framework* está compuesto de tres capas y dos extensiones, cada una de ellas conformados por un grupo de componentes que al relacionarse generan vistas de arquitectura. Cada una de las capas está compuesta por diversos elementos, los cuales ayudan al arquitecto a realizar los diseños de las dimensiones arquitectónicas de Togaf. La primera capa cubre el proceso de la empresa, permite visualizar como los procesos, objetos de negocio, los roles, y otros componentes se relacionan para prestar servicios y ofrecer productos a los clientes. La segunda capa permite modelar los datos y aplicaciones, generando herramientas para visualizar como los servicios de aplicaciones apoyan a los procesos de negocio u otros sistemas de información, relacionan los artefactos u objetos de negocio con las entidades de dominio, detalla los módulos de un sistema de información, entre otros. Y por último está la capa de infraestructura que ayuda a visualizar el *software*, *hardware* y redes que soporta los sistemas de información.

The Open Group (2013) Clasifica los elementos en tres categorías, los activos son aquellos que inician una acción, por ejemplo, los actores de negocio, componentes de aplicaciones, dispositivos, entre otros. Los elementos de comportamiento son los que generan una transformación y son iniciados por los elementos activos, los elementos de comportamiento son los procesos, funciones, eventos y servicios. Por último, los pasivos son aquellos afectados por la acción u objetos de comportamiento, los componentes que representan los objetos pasivos son los objetos de datos u objetos físicos.

The Open Group (2013) y Open Group (2016) Explican que la extensión de estrategia y motivación cubre las fases A, H, preliminar y requisitos, permitiendo al arquitecto modelar aspectos tales como objetivos, interesados, principios, requerimientos, restricciones, capacidades, acciones en curso, recursos, entre otros. Por otra parte, la extensión de implementación y migración cubre las fases E, F, G y H del ADM de Togaf, usando los componentes de entregable, paquete de trabajo, brecha y mesetas

(arquitectura de transición), con dichos componentes el arquitecto puede modelar el mapa de ruta, las diferentes arquitecturas de transición y las brechas identificadas entre las mismas.

Carlos L.B. Azevedo (2015) hace énfasis en el modelado de recursos y capacidades, las cuales las propone usando el lenguaje Archimate. Resalta que los recursos de la organización y las capacidades son un factor clave para el éxito de una compañía, por tal motivo la evaluación de las capacidades de organización y los recursos deben enfocarse en mejorar el rendimiento, la calidad y reducir los costes.

Con la siguiente imagen se ilustra como Archimate 3.0 cubre cada una de las etapas del ADM de Togaf 9.1

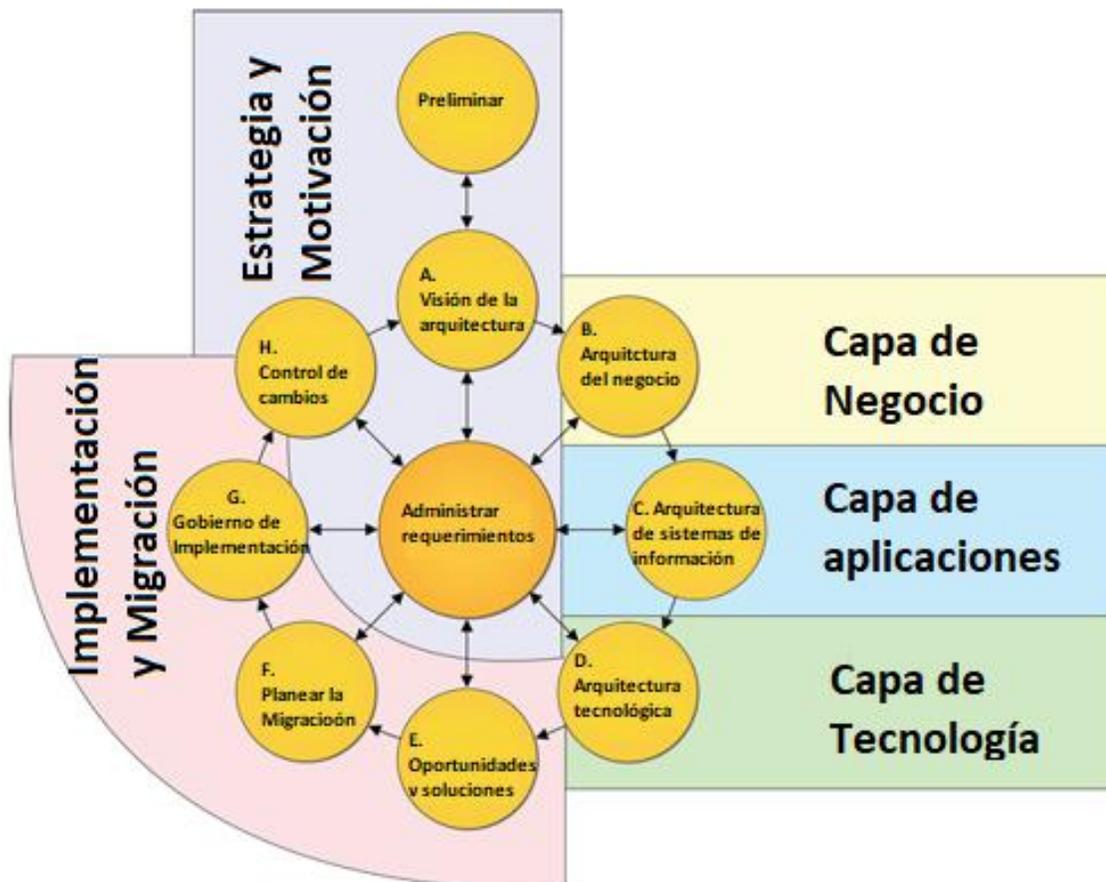


Ilustración 3. Relación entre Togaf 9.1 y Archimate 3.0 ³

³ Tomado de (Open Group, 2016) – Imagen traducida por la autora del trabajo de grado

4.2.2.2. *Unified Model Lenguaje (UML)*

Según Weillkiens & Oestereich (2007) el modelo de lenguaje unificado o UML es un lenguaje compuesto por un grupo de diagramas que permiten documentar diversos modelos de *software*. La variedad de diagramas que compone UML permite modelar diferentes vistas, como son el diseño, la construcción, el despliegue, entre otros.

Weillkiens & Oestereich (2007) afirma que Booch, Rumbaugh y Jacobson, crearon la primera versión de UML y en 1997 el lenguaje de modelado fue aprobado por el *Object Management Group* (OMG), desde ese momento UML ha venido evolucionando, con el fin de fortalecer y enriquecer los diagramas que lo componen, actualmente UML está en la versión 2.0.

Los diagramas están clasificados en estructurados y de comportamiento, Pilone & Pitman (2005) expone las características de cada grupo. Los estructurados son llamados así por su capacidad de capturar la organización física al interior de un sistema. Este grupo está compuesto por 6 diagramas:

- *Diagrama de paquetes*
Spark System (2015) Resume el diagrama de paquetes como una agrupación de conceptos o elementos organizados para representar relaciones físicas y lógicas. El diagrama de paquetes ayuda a organizar sistemas complejos, agrupándolos por un criterio particular y facilitando su modelamiento y entendimiento.
- *Diagrama de clases*
Spark System (2015) Describe el diagrama de clases como un modelo de bloques de construcción de un sistema orientado a objetos, las clases están compuestas de atributos y funciones, estas últimas representan los comportamientos que puede tener dicha clase. Adicionalmente permite modelar las relaciones y el tipo de relación entre las clases y las interfaces del sistema de información.
- *Diagrama de objetos*
Según Pilone & Pitman (2005) el diagrama de objetos tiene la misma sintaxis que los diagramas de clases, la diferencia entre ambos sistemas, consiste en que el diagrama de objetos muestra la relación entre objetos en tiempo de ejecución y su comportamiento.

- *Diagrama de estructuras compuestas*
Spark System (2015) Define el diagrama de estructura compuesta como un diagrama que muestra la relación y puntos de interacción entre clases y partes, formando así estructuras más complejas y funcionales.
- *Diagrama de componentes*
Según Pilone & Pitman (2005) el diagrama de componentes permite ver la estructura y dependencia de las partes en la implementación de un sistema. Cada componente representa el conjunto de objetos pequeños de una forma desplegable e instalable. El diagrama puede ser modelado a un alto nivel (instalador) o detallado (funciones principales) según las necesidades del proyecto.
- *Diagrama de despliegue*
Kimmel (2008) describe el diagrama de distribución o despliegue como a forma en que un sistema debe quedar almacenado cuando se esté usando (independiente del ambiente). Muestra los elementos lógicos en sus ubicaciones físicas y cómo se van a comunicar entre ellos. Permite visualizar la tecnología y plataformas reales con las cuales cuenta una empresa. Pilone & Pitman (2005) afirma que el diagrama de despliegue muestra como un sistema se está ejecutando físicamente, es decir, permite visualizar en que servidores está alojado que bases de datos consume, estas bases de datos dónde están ubicadas, si están en clúster, entre otros.
- *Modelo de dominio*
Larman (2003) Define el modelo de dominio como una representación visual de las clases conceptuales o los objetos del mundo real, compuesto de atributos y relaciones entre entidades, también es conocido como modelo conceptual.
- *Modelo entidad relación*
Barker (1990) Define el modelo entidad relación como una técnica usada para desarrollar un modelo de datos de alta calidad, ya que permite definir los datos y las relaciones que existen entre las tablas, adicionalmente es un lenguaje universal que permite el entendimiento por parte de todos los interesados.
- *Diagrama de capas*
Microsoft (2016) define el diagrama de capas como un agrupador lógico de artefactos físicos el sistema, las capas describen las principales tareas que realizan los componentes principales del sistema. El diagrama de capas es usado para: Comunicar arquitectura lógica, detectar conflictos entre el código existente y la arquitectura planeada, revisar los impactos generadas a la arquitectura por posibles cambios y tener un soporte arquitectónico para el mantenimiento.

Los diagramas de comportamiento reflejan los cambios de los componentes al interior de un sistema, por ejemplo, cómo cambia un objeto de un estado al otro y cuál es el disparador que inicia el cambio. Este grupo está compuesto por 7 diagramas

- Diagrama de casos de uso
Según Pilone & Pitman (2005) y Spark System (2015) el diagrama de casos de uso permite visualizar la interacción que tiene un usuario con el sistema de información y a su vez capturar los requisitos funcionales del *software*.
- Diagrama de actividades
Spark System (2015) Describe el diagrama de actividades como un diagrama que permite ver el flujo desde el inicio hasta el fin de las mismas y la relación entre componentes, adicionalmente se usa para detallar flujos paralelos de los procesos.
- Diagramas de máquinas de estado
Según Pilone & Pitman (2005) y Spark System (2015) el diagrama de máquina de estados o diagrama de estados modela los eventos que un objeto en particular puede presentar durante su tiempo de vida
- Diagramas de comunicación
Pilone & Pitman (2005) Define los diagramas de comunicación como una interacción entre los objetos de un sistema. La información de las relaciones entre los componentes es muy enriquecida, ya que cuenta con mensajes y enumeraciones que representan la secuencia. En versiones anteriores el diagrama se conocía como un diagrama de colaboración.
- Diagramas de secuencia
Según Spark System (2015) el diagrama de secuencias muestra la interacción entre los componentes mediante el orden de ejecución (Algunos pueden ser paralelos y evidencias ejecución asincrónicas), los mensajes que son enviados y recibidos entre objetos durante la ejecución.
- Diagramas de tiempos
Según Spark System (2015) el diagrama de tiempo se usa para mostrar el comportamiento o cambios de los elementos en el tiempo. Adicionalmente puede mostrar restricciones en el tiempo, duración máxima permitida de eventos, entre otros.

- *Diagrama de interacción*
Según Spark System (2015) el diagrama de descripción de la interacción es similar a un diagrama de actividades, la diferencia consiste en que las actividades son representadas por diagramas de interacción como son los diagramas de comunicación, secuencia, descripción de la interacción y tiempos.

4.2.2.3. Business Process Modeling Notation (BPMN)

Antes de empezar a hablar del lenguaje de modelado BPMN 2.0 es importante definir que es BPM y qué es un proceso. Hitpass, (2017) define BPM como “*Disciplina de Gestión por procesos de Negocio y de Mejora Continua apoyada fuertemente por las Tecnologías de la Información*”. Lo anterior se traduce en el análisis y gestión de procesos que ayudan a alcanzar los objetivos estratégicos de una organización, teniendo como base la relación de personas, sistemas de información, negocios, clientes, proveedores y socios.

Weilkiens, Weiss, & Grass, (2011) define un proceso de negocio como una serie de pasos diseñados para producir un producto o servicio. Si el resultado generado beneficia directamente el cliente externo se conoce como un proceso primario, de lo contrario es un proceso de apoyo. La descripción de los procesos y el modelado de estos se puede realizar usando el lenguaje de modelado BPMN.

Weilkiens, Weiss, & Grass, (2011) y Freund, Rücker, & Hitpass (2014) exponen que la primera versión de BPMN fue desarrollada por el Instituto de administración de procesos de negocio BPMI, con el liderazgo de Stephan A. White quien era en ese entonces un profesional de IBM. El objetivo principal de la creación del lenguaje era tener una base gráfica para automatizar los procesos. En el 2005 el proyecto fue trasladado a Object Management Group (OMG) porque este instituto tenía una amplia experiencia en la administración y publicación de estándares. Actualmente la versión de BPMN es la 2.0 publicada en el año 2010

Según Pais (2013) el BPMS o Notación de modelado de procesos es un lenguaje que permite estandarizar los diagramas que modelan los procesos de una compañía. A partir de los modelos todos los interesados pueden visualizar los flujos de procesos, identificar las interacciones entre áreas y artefactos importantes que hacen parte del proceso. PBMN es un lenguaje

sencillo y flexible, permitiendo diagramar modelos a alto nivel o muy detallados, según lo requiera la organización.

Pais (2013) y Weilkiens & Oestereich (2007) coinciden en que BPMN no es el objetivo final de la administración de negocio, al contrario, es un paso para llegar al objetivo que es la automatización de los procesos mediante herramientas de TI. La automatización permite enlazar cada una de las actividades que hacen parte del BPMN a funciones o servicios desarrollados en plataformas tecnológicas.

Teniendo en cuenta que el modelado BPMN es la primera etapa del BPM Michele Chinosi (2012) hace énfasis en que los *Business Process Execution Language* (BPEL) puede considerarse la mejor opción para traducir diagramas BPMN en código ejecutable directamente. Shi (2016) define los BPEL como un lenguaje basado en XML para especificar los procesos de negocio como servicios, adicionalmente con su descripción de la composición de servicios Web. El uso de BPEL se refleja en la integración de aplicaciones empresariales. Cabe resaltar que como BPEL no gráfico, gerentes y analistas de negocios tienen problemas para comprender los códigos. Michele Chinosi (2012) resalta la importancia de los BPEL ya que tiene amplios mecanismos para tratar las excepciones del proceso de negocio, define como las actividades deben ser compensadas en algunos escenarios, ofrece esquemas en XML y son diseñado para almacenar los aspectos de un diagrama BPMN.

Weilkiens, Weiss, & Grass, (2011) habla del estándar de la OMG conocido como modelo de motivación del negocio (BMM) que se enfoca en los objetivos, visión, estrategia y tácticas que una empresa define. Se resalta este tema en el trabajo por la similitud que tiene con Togaf con la fase del ADM de definición de la visión.

Los tópicos o temas en los que se enfoca BMM con los siguientes:

- Fin: Describe la visión, metas y objetivos de la empresa y la relación entre estos componentes
- Significado: Se refiere a la misión, estrategias y tácticas.
- Factor de influencia: Describe las influencias externas que podrían cambiar el rumbo o estrategia de la empresa.
- Evaluación: Evalúa los factores internos y externos de la empresa para determinar si el rumbo es el correcto.
- La información externa: Es la integración de toda la información de la empresa con BMM. Por ejemplo: la definición de procesos.

Harmon (2007) y OMG (2016) coinciden en que BPMN 2.0 soportan un conjunto básico de elementos para entender y comunicar los procedimientos internos de negocio, adicionalmente resalta que los arquitectos de procesos modelan para simplificar, resaltar, aclarar y comunicarse, lo que exige que los diagramas sean de fácil entendimiento. Hacen énfasis en que el conjunto básico de elementos de notación BPMN proporciona los elementos claves para lograr la comprensión de todos los colaboradores al interior de una organización, asegurando que las empresas entiendan las interacciones que tiene al interior de ella misma o con otras empresas (B2B).

4.2.3. Elicitación de requisitos

Los métodos o estrategias seleccionadas para identificar los requisitos del proyecto deben abarcar los requisitos funcionales y los atributos de calidad, logrando así que las arquitecturas de procesos, datos, sistemas de información y tecnología cumplan con las expectativas de los interesados.

Nuseibeh & Easterbrook (2000) Clasifica las técnicas de elicitación de requisitos en seis categorías, la elección de aplicar alguna de las técnicas depende del tiempo, recursos disponibles, tipo de información y experiencia del ingeniero de requisitos. Las categorías son:

- Las técnicas tradicionales comprenden técnicas como encuestas, entrevistas, análisis de documentación y cuestionarios
- Las técnicas grupales intentan promover el consenso y la participación de los interesados. En esta categoría se encuentra la lluvia de ideas, *workshop* y los grupos focales.
- La técnica de prototipos es usada cuando la incertidumbre es alta y se requiere que los interesados visualicen como quedarán implementados los requerimientos.
- La técnica dirigida por modelos o *Model Drive* incluyen métodos como los escenarios y el estudio de los objetivos, con el fin de entender la organización.
- Las técnicas cognitivas comprenden el análisis de protocolo, escalamiento, tarjeta de la clasificación, rejillas de repertorio.
- Técnicas contextuales nacieron en los 90's como una alternativa a las técnicas tradicionales y cognitivas. Estos incluyen el uso de técnicas

etnográficas como la observación del participante y el análisis de conversaciones.

Dante Carrizo (2014) expone que la selección de una técnica para la obtención de requisitos tiene un proceso sistémico que se compone de tres pasos. El primer paso es la identificación del contexto actual, que ayuda a identificar las características del contexto del proyecto e indica cuáles son las técnicas más adecuadas para la elicitación. El segundo paso es adecuar las técnicas a la situación, que consiste en evaluar la idoneidad de cada técnica en el contexto del problema. El tercer y último paso es obtener un plan de sesión, donde se priorizan y seleccionan las técnicas identificadas anteriormente y programan para cada sesión.

Cabe resaltar que los atributos de calidad o requisitos no funcionales juegan un papel muy importante en la elicitación de requisitos, ya que son estos los que dan los lineamientos para el diseño de la arquitectura y garantizar que las expectativas del cliente se cumplan. *Software Engineering Institute - SEI* (2015) Desarrolló un taller para la identificar los atributos de calidad, donde se involucran diferentes interesados, tanto técnicos como funcionales. El objetivo principal del taller es lograr que los integrantes manifiesten sus necesidades de usabilidad, disponibilidad, escalabilidad, seguridad, desempeño, entre otros, mediante escenarios. El taller es conocido como *Quality Attribute Workshops o QAW*.

Mario R. Barbacci (2003) hace énfasis en que el método es usado para generar, priorizar y refinar los escenarios de atributos de calidad antes de que se complete la arquitectura de *software*. En este taller están todas las partes interesadas que tiene relación con el sistema de información que se va a desarrollar, los participantes se componen de los directivos, usuarios funcionales y personas de tecnología. La contribución de cada una de las partes es fundamental ya tienen diferentes puntos de vista que ayudan a tener un panorama completo de las expectativas y restricciones del sistema de información a desarrollar. El QAW cubre los siguientes pasos:

- Presentación e introducción QAW
- Presentación del negocio y la misión de la empresa o procesos afectados
- Presentación de la arquitectura del negocio
- Identificación de los conductores (*drivers*) de arquitectura
- Lluvia de escenario
- Consolidación escenario
- Priorización escenario
- Escenario refinamiento

4.3. Estado del arte

Las investigaciones y trabajos citados en el estado del arte se relacionan directamente con el marco conceptual, ya que son la base teórica para el trabajo de investigación que se está proponiendo. El estado del arte se divide en dos grandes bloques como son la arquitectura empresarial y la técnica de elicitación de requisitos, ya que son temas fuertemente desarrollados e investigado por otros autores. Cabe resaltar que las investigaciones citadas están relacionadas con las necesidades estratégicas de la compañía y cómo son resueltas con el análisis de procesos y el desarrollo de *software* empresarial.

4.3.1. Arquitectura empresarial

La información consignada mediante trabajos académicos de investigación o de implementación de forma general tratan los puntos clave del proyecto de manera individualizada. Algunos utilizan las notaciones o las describen en sus marcos referenciales, pero es escasa la integración de todos los elementos en un artículo, de tal manera que los avances en algunos casos involucran algunos de manera conjunta. Como lo describe Nasrin Dastranj Mamaghani (2012) actualmente existen diferentes modelos de referencia de arquitectura empresarial y los marcos para el diseño y desarrollo de sistemas de información para los cuales cada empresa adecua o toma lo necesario según sus necesidades, para no realizar esfuerzos en elementos que finalmente no aportarán al diseño, desarrollo, implementación y mantenimiento en el tiempo como lo expone en sus preguntas claves para responder las dimensiones clave de empresa.

En un comparativo entre herramientas BPMN Mesa , Lochmuller, & Tabares, (2014) hace especial énfasis que debe existir un buen manejo en los procesos del negocio, pero para la automatización de ellos es indispensable contar con herramientas de arquitecturas de negocio para adaptar el proceso a diferentes servicios informáticos. Se expone una propuesta *“Para el trabajo futuro se espera analizar las herramientas a nivel de ejecución y monitoreo de los procesos de negocio con el fin de entender mejor la relación que existe entre los modelos de procesos de negocio bajo el estándar BPMN, el proceso de desarrollo de software, el apoyo de éstas a la arquitectura de negocio definida por TOGAF y arquitecturas enfocadas a servicios”* (Mesa , Lochmuller, & Tabares, 2014).

En el trabajo de Kurpjuweit & Winter (2009) se sustenta que las organizaciones están sujetas a una evolución constante y deben analizar los impactos de los cambios de manera sistémica y aplicarse de forma coherente en todos los dominios de la organización, es importante mantener el conocimiento en detalle de los artefactos y sus relaciones. La gestión de la arquitectura empresarial es una vía para mantener y asegurar la correcta actualización de la documentación de estos artefactos. En su propuesta intenta la generación de algunas plantillas de arquitectura empresarial como bases para las metas de referencias de los modelos propuestos para aplicarlos a los requisitos específicos de la empresa.

Las experiencias de los autores en los artículos referenciados muestran la importancia de combinar los diferentes elementos para concretar una metodología que oriente el uso de los artefactos de diseño de los elementos clave en el mantenimiento a lo largo del tiempo.

Los artefactos de diseño deben confluir y contrastar con los procesos de negocio, garantizando una visión global de las necesidades de la empresa, con el fin de generar reutilización, optimización de tiempos y uso de los elementos de cada sistema de información desarrollado a la medida o de las adecuaciones que se realizan para garantizar el flujo de información entre los procesos de la compañía con una visión macro apoyada en los elementos de arquitectura empresarial, de igual forma se deben alinear los componentes tecnológicos que comprenden las soluciones que atienden los sistemas de información.

Esta información es clave para tratar de enfocar las inversiones en tecnología que no vayan en detrimento de los intereses y optimice el uso de la información y maximice el tiempo requerido para que los flujos de atención en la compañía se realicen en la oportunidad deseada.

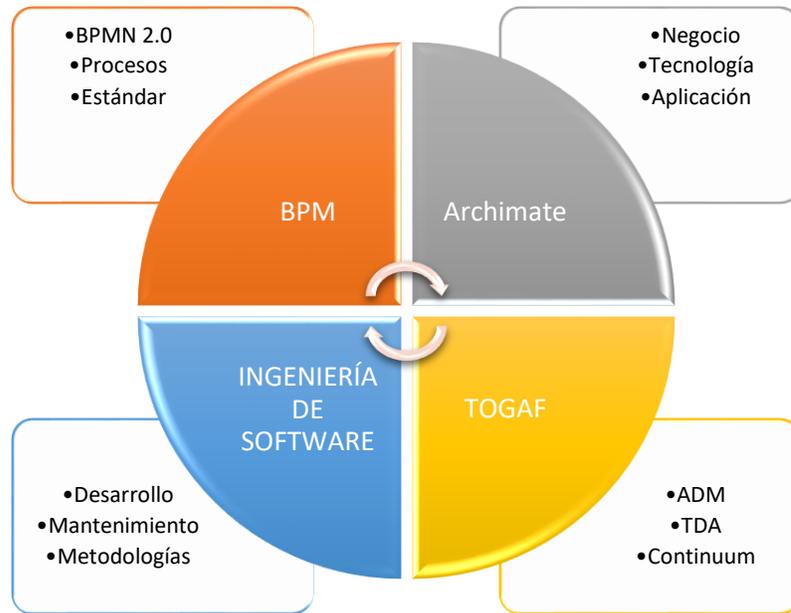


Ilustración 4. Relación entre BPM, Archimate, Ingeniería de software y Togaf ⁴

La Ilustración 5 muestra cómo han cambiado los conceptos. “Anteriormente la arquitectura empresarial contaba con el soporte de sistemas separados independientes. Había un discreto distanciamiento entre la arquitectura empresarial y los ingenieros de sistemas, con sus problemas afines. Hoy en día el desarrollo de sistemas se basa mucho más en los negocios. Hay una fuerte necesidad de responsabilidad financiera de TI y los gastos en sistemas y éstos deben ajustarse a su beneficio comercial. Por ello, la alineación entre negocio y desarrollo es crucial. Hay una participación constante entre el arquitecto empresarial y el ingeniero de sistemas que conduce a una mayor alineación negocio/TI y colabora con la gobernabilidad técnica en todo el ciclo de vida, los arquitectos empresariales participan durante más tiempo y los ingenieros de sistemas se involucran antes. Por último, los servicios implementados soportan la obtención y el monitoreo de datos durante la operación. El análisis de este intercambio impulsa cambios futuros”. (Dave Brown, 2015)

⁴ Gráfico generado por el autor del trabajo de grado

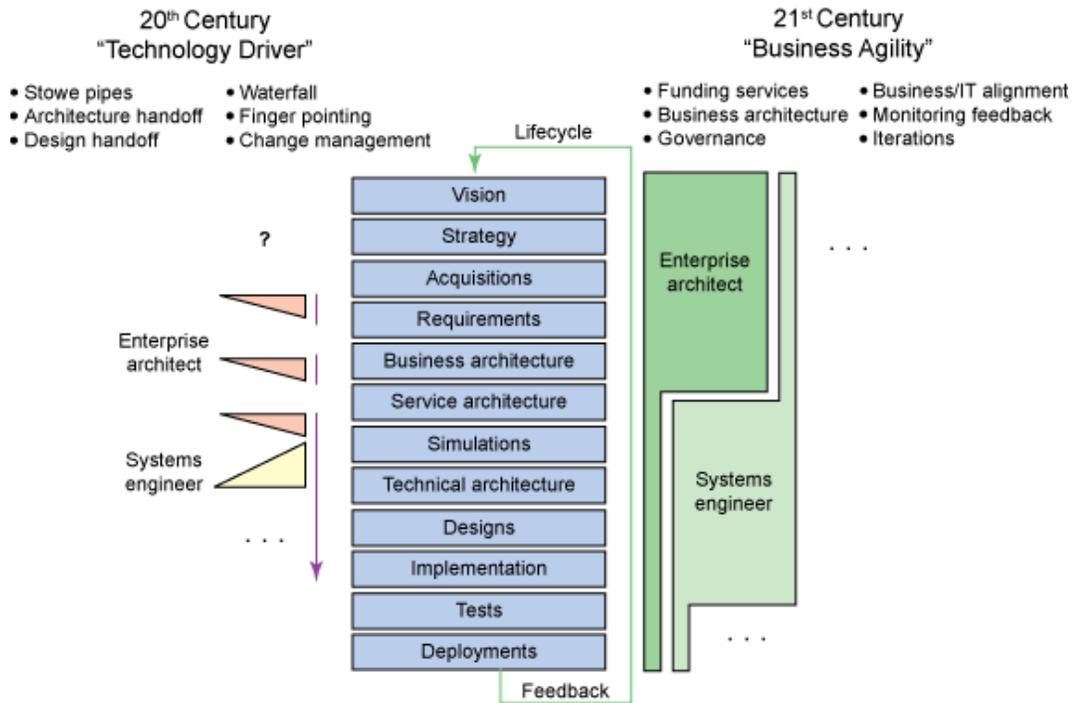


Ilustración 5. Alineación de arquitectura e ingeniería al inicio del ciclo de vida⁵

No todos los autores apoyan marcos de trabajo para implementar en la empresa por su robustez y dificultad. (Ernst, 2008) en su artículo habla de la creación de tres tipos de patrones a raíz de lo complejo que era la implementación de grandes *frameworks* como Zachman y Togaf al interior de una organización.

Se presentan tres tipos diferentes de patrones, los *M-Patterns* hacen frente a los problemas típicos en la gestión de arquitectura empresarial. *V-Patterns* representan a las visualizaciones de mejores prácticas, mientras que *I-Patterns* indican los requisitos de información para la gestión de la arquitectura empresarial.

Los *M-Patterns* o patrones de metodología abordan los problemas de gestión de una manera escalonada, es decir especifican actividades que se deben llevar a cabo en el proceso de gestión. A diferencia de TOGAF o marcos de trabajo robustos que proporcionan modelos de proceso, pero dejan los detalles de las metodologías de apoyo abiertas, es decir, cada compañía debe buscar la forma de materializarlas.

⁵ Tomada de: (The Rational Edge, 2015)

Los *V-Patterns* o Patrones de punto de vista proporcionan visualizaciones basadas en diagramas e informes que permiten a los interesados interpretar las vistas desarrolladas según su interés. Con el fin de asegurar el entendimiento de una vista según un punto de vista, una leyenda o descripción debe ser obligatorio. Los *V-Patterns* se pueden utilizar como una utilidad por uno o más *M-Patterns*.

Por último, están los *I-Patterns* o patrones de información, los cuales suministran mejores prácticas en los modelos de información, incluyendo las definiciones y descripciones de los conceptos utilizados. La información puede ser visualizada en las vistas de acuerdo con una o más *V-Patterns* o ser utilizada directamente por *M-Patterns*. Permite el uso de lenguajes de modelado como son: el Fondo para el Meta Objeto (MOF), *Unified Modeling Language* (UML) y formalizaciones matemáticas o combinaciones de idiomas. La elección de un lenguaje específico tiene que tener en cuenta las necesidades que se deben apoyarse.

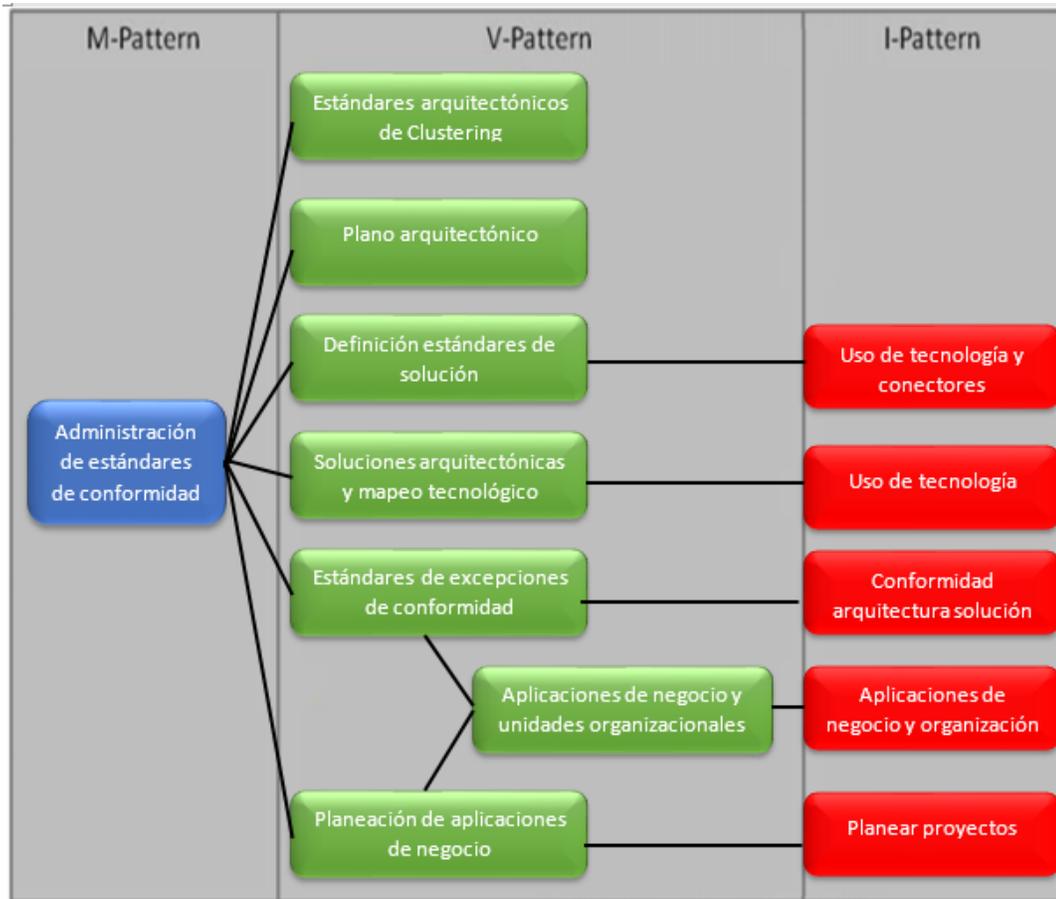


Ilustración 6. Mapa de patrones para el artículo ⁶

⁶ (Ernst, 2008) – Traducción realizada por la autora del trabajo de grado

En otras investigaciones Napoli & Kaloyanova (2011) hace una investigación para tratar de validar si la metodología RUP puede adoptar principios complementarios, funciones y mecanismos de otras metodologías complementarias a la ingeniería de *software*, como son la arquitectura empresarial, arquitectura orientada a servicios y gestión de procesos de negocio. El trabajo se basa en el análisis y la extracción de los componentes TOGAF, SOA y BPM que se pueden asignar en el ciclo de vida RUP iterativo. Concluyen a la investigación lo siguiente:

- Es importante contar con procesos BPM y metodologías para fortalecer el ciclo de vida del gobierno de la empresa, con el fin de asegurar la continuidad del modelo de negocios.
- Un modelo de programación SOA debe garantizar la calidad de servicio, los acuerdos de niveles de servicio y un gobierno que garantice el ciclo de vida de la aplicación de *software*, con el fin de generar una retroalimentación útil para tener un mejoramiento continuo
- Existe sinergias en los temas analizados, se cree que todos los factores convergen para garantizar una automatización, calidad de servicio, niveles de servicio, y cumplimiento del gobierno.

Hinkelmann (2015) Presentan un caso de estudio enfocado en empresas ágiles que requieren interacción de personas, información y tecnología. Su investigación está basada en identificar el desafío para las empresas para alinear continuamente el negocio con las tecnologías de información en un entorno que cambia rápidamente.

Cuando habla de empresas ágiles se refiere a aquellas compañías que se adaptan rápidamente a nuevos desafíos y oportunidades de negocio con el fin de optimizar los costos, calidad y desempeño. Todos estos cambios van de la mano de los procesos, los sistemas de información y las tecnologías.

Con el fin de hacer frente a este reto Hinkelmann (2015) propone un enfoque que se basa en los principios del modelo impulsado por la empresa de ingeniería y se complementa con dos desarrollos innovadores y desafiantes: cambiar el paradigma de modelos de ingeniería por desarrollos para adaptación continua y contar con el apoyo de las máquinas, sin embargo los humanos interpretando los modelos, es decir, los modelos no deben quedar de forma pasiva en una máquina, el humano debe entenderlos para evolucionarlos. Como resultado se desarrolló un metamodelo que se basa en

la ingeniería del conocimiento para la gestión de procesos de negocio. Las características del metamodelo son:

- Notaciones gráficas, fáciles de comprender por los seres humanos
- La semántica del metamodelo permita hacer el análisis, adaptación y evaluación del modelo por una máquina

4.3.2. Técnicas de elicitación de requisitos

En esta sección se listan varios estudios e investigaciones desarrolladas por diversos autores, donde se exponen sus propuestas y la estructuración de estas.

En el estudio realizado por Wellsandt, Hribernik, & Dieter T (2014), se identificaron ocho técnicas de elicitación de requisitos, basadas en seis criterios para seleccionar las mejores y más eficientes para cada proyecto.

Las técnicas identificadas por Wellsandt, Hribernik, & Dieter T (2014) son las siguientes:

- Cuestionarios (I): son preguntas diseñadas con anterioridad las cuales se pueden presentar por diferentes medios, ya sean virtuales o escritas. Las preguntas pueden ser abiertas o cerradas según se identifiquen las necesidades del proyecto.
- Entrevistas (II): son menos formales que los cuestionarios y más personalizadas, ya que se está con la persona cara a cara. Las preguntas deben prepararse con antelación, estas pueden ser abiertas o cerradas y es el entrevistador quien guía la reunión.
- Informes de usuarios (III): Esta no es una técnica muy confiable, ya que los informes presentados por los usuarios están sujetos a su forma de trabajar y pueden ser muy subjetivos.
- En el sitio del cliente (IV): Esta técnica implica visitas del cliente a las instalaciones de desarrollo, se busca tener una retroalimentación a corto plazo del proyecto que se está desarrollando. Esta técnica requiere esfuerzos mayores para involucrar al usuario y generalmente se incluye un número muy limitado de personas.
- Observación en campo (V): En esta técnica la persona encargada de tomar los requisitos se encuentra en campo observando cómo se ejecutan los

procesos involucrados. Pueden apoyarse de cámaras y grabadoras para dejar material para consultas futuras.

- Aprendizaje (VI): Basados en la observación y la práctica se adquiere una comprensión más clara y profunda del proceso y las necesidades del producto.
- Sensores de productos embebidos (VII): Similar a la observación en campo, pero en este caso no son personas observando sino sensores incrustados recopilando información sobre el producto de forma remota.
- Gestión de reclamaciones (VIII): Las quejas o inconformidades de los clientes es información valiosa para el análisis y mejora de los productos y necesidades de los usuarios. Con la información recopilada se pueden mejorar los procesos de negocio.

Los criterios de evaluación son los siguientes:

- Proximidad al uso (PTU): Describe el intervalo de tiempo entre el uso real y la aplicación correspondiente a la técnica de elicitación.
- Esfuerzo por usuario (EPU): Se evalúa todo el esfuerzo que debe tener una persona para la ejecución de la técnica de elicitación de requisitos, adicionalmente se debe tener en cuenta los costos reflejados en el tiempo y el material requerido.
- Habilidades requeridas (SKI): Habilidades o grado mínimo de conocimiento que debe tener una persona para ejecutar la técnica de elicitación de requisitos.
- Cuantificable (QUA): Valida si la información se puede medir, ya que la información cuantificable tiene un menor riesgo a errores y es más fácil de procesar.
- Estructura (STR): Mide las irregularidades o ambigüedades en las cuales se incurre dependiendo de la técnica de elicitación de requisitos.
- Riqueza (RIC): La riqueza de la información se mide a partir de la interpretación de los datos, es decir, si una persona expresa su necesidad es más fácil de comprender, pero si se entregan números sin contexto solo se considera información plana.

Wellsandt, Hribernik, & Dieter T (2014) concluye en la investigación con la presentación de una matriz que incluye las técnicas y los criterios de evaluación.

Criterios	Técnica no.							
	I	II	III	IV	V	VI	VII	VIII
ptu	m	m	h	m	h	h	h	l
epu	l/ma	m/few	l/ma	h/few	h/few	h/few	m/ma	h/few
ski	l	m	l	m	h	m	m	l
qua	m	m	m	m	m	m	h	m
str	mo	le	mo	le	mo	le	mo	mo
ric	m	h	l	h	m	h	l	m

I = low | m =medium | h = high | ma = many | mo = more | le = less

Ilustración 7. Resultado de la evaluación cualitativa y las técnicas de elicitación de requisitos⁷

Wüffel, Lutz, & Diehl (2015) afirma que la elicitación de requisitos es un proceso fundamental para el desarrollo de *software*, ya que en esta actividad se define el alcance de la solución. Wüffel, Lutz, & Diehl (2015) ha desarrollado una metodología conocida como *Grounded Requirements Engineering* (GRE) que consiste en observar y analizar los procesos y actividades de los usuarios en el mundo real.

GRE es un proceso iterativo compuesto de dos partes. La primera actividad está basada en los métodos de Teoría Fundamentada, los cuales son usados para analizar la información capturada mediante entrevistas a los usuarios. La segunda actividad consiste en analizar el comportamiento de los usuarios y convertirlos en casos de uso. GRE produce requisitos comprensibles y conservan la trazabilidad desde el proceso hasta los sistemas de información.

Salgado, Machado, & Suzana (2013) consideran que la ingeniería de Requisitos juega un papel central en la especificación del sistema de información y la alineación entre el negocio y las tecnologías de información. Presentan un trabajo en la obtención y gestión de requisitos en contextos mal definidos, presenta algunos retos que deben abordarse desde otras perspectivas como son el modelo de negocio, estrategia y objetivos enfocados en la arquitectura empresarial. Desarrollan BISITA que es un modelo que abarca la estrategia, los objetivos y la estructura del negocio, con el objetivo

⁷ (Wellsandt, Hribernik, & Dieter T, 2014)

de generar sinergia entre la empresa y los objetivos estratégicos de las tecnologías de información. Se concluye que un modelo de negocio puede ser utilizado para ingeniería de requisitos mediante la recopilación y representación de los objetivos de negocio a un alto nivel. En relación con el modelado de empresa, se identifica un excelente trabajo de integración y alineación de negocio y de tecnologías de información usando como lenguaje de modelado ArchiMate y marco de trabajo TOGAF permitiendo así una trazabilidad de la estrategia con los objetivos y requisitos.

5. METODOLOGÍA

El proyecto se va a desarrollar sobre una metodología cualitativa, ya que la base de la investigación no serán datos estadísticos, al contrario, se basa en observaciones y análisis de situaciones reales que se presentan día a día en las compañías.

Las fases a trabajar en el proyecto son las expuestas por Quintana (2006):

Formulación

En la formulación se identifica el problema y el alcance del proyecto, adicionalmente los temas e información fundamental para el desarrollo del proyecto.

Diseño

En el diseño realizará una investigación Togaf 9.1, Archimate 3.0, UML 2.0, BPMN 2.0 y elicitación de requisitos y posteriormente se analizará la información para extraer los elementos que aportan valor al trabajo de grado y ser aplicada en un marco de trabajo adaptado para el desarrollo de *software* empresarial.

Adicionalmente se dejarán blogs con información clara y accesible para las personas que quieran conocer más del tema e incluso aportar experiencias y conocimientos.

Ejecución

En la ejecución se realizará un piloto para aplicar la metodología sobre un proyecto de desarrollo de *software* empresarial, con el fin de identificar fortalezas y oportunidades del marco de trabajo adaptado.

Cierre

En el cierre se entregará el trabajo terminado con las investigaciones realizadas, el marco de trabajo consolidado y los resultados y conclusiones de la prueba piloto aplicada a un desarrollo de *software* empresarial.

6. PRESENTACIÓN Y ANÁLISIS DE RESULTADOS

El desarrollo del proyecto se dividió en cuatro objetivos, a continuación, se detallará el desarrollo de cada uno

6.1. Objetivo 1 - Documentación de apoyo de la metodología

Se resalta que esta documentación es la base para que las personas tengan una herramienta de autoaprendizaje, donde pueden encontrar información redactada de forma sencilla, clara, concisa y en español.

En esta sección se tendrá detallado cada tema que soporta el marco de trabajo adaptado. Para efectos del trabajo de grado esta información quedará en el documento actual como una línea base, sin embargo, se entregará una wiki con la información de apoyo con el fin de que esta sea consultada por los integrantes de los proyectos de arquitectura empresarial y puedan enriquecerla con las lecciones aprendidas, actualizaciones o modificaciones de las metodologías y lenguajes que apoyan el marco de trabajo adaptado.

WikiIUPB [Discusión](#) [Preferencias](#) [Lista de seguimiento](#) [Contribuciones](#) [Salir](#)

Página [Discusión](#) [Leer](#) [Editar](#) [Ver historial](#) [Más](#)

Marco de trabajo adaptado para proyectos de desarrollo de software empresariales usando Togaf 9.1

Este trabajo se soporta en la investigación del marco de trabajo Togaf 9.1 y los lenguajes de modelados Archimate 3.0, UML 2.0 y BPMN 2.0 con el fin de adaptar Togaf 9.1 a los proyectos de desarrollo de software empresariales y garantizar la alineación de estos con la estrategia de la compañía. Dicha alineación se realizará mediante talleres, herramienta, formatos y una estructura de documentación que agrupe los lenguajes de modelado mencionados anteriormente.

A partir de los entregables definidos se podrá diagramar el modelo motivacional, las arquitecturas de negocio, aplicaciones, datos e infraestructura y la relación entre ellas, adicionalmente se podrán visualizar brechas, capacidades arquitectónicas de la organización y gobierno arquitectónico.

La documentación que contiene este sitio se divide en tres partes, la primera de ellas es la documentación de apoyo, donde encuentra la teoría de la metodología, es decir, los integrantes del equipo podrán conocer los lenguajes de modelado y teoría base para el entendimiento de la metodología propuesta. La segunda parte se enfoca en cómo aplicar los talleres y diligenciamiento de formatos en cada una de las fases del ADM, es decir, se centra en la metodología propuesta y la tercera y última parte especifica las arquitecturas de referencia para diferentes tipos de proyecto, con el fin de que los arquitectos y desarrolladores tengan diseños predeterminados y orientados a las mejores prácticas.

Documentos de apoyo	Documentos de la metodología	Arquitecturas de referencia
<ul style="list-style-type: none">Documentos de Apoyo<ul style="list-style-type: none">Archimate 3.0BPMN 2.0Elicitación RequisitosScrumTogaf 9.1UML 2.0	<ul style="list-style-type: none">Documentos de Metodología<ul style="list-style-type: none">Fase A - Visión de la arquitecturaFase B - Arquitectura del negocioFase C - Arquitectura de sistemas de informaciónFase D - Arquitectura tecnológicaFase E - Oportunidades y solucionesFase F - Planificación de la migraciónFase G - Gobierno de la implementaciónFase H - Gestión del cambio de los requisitos	<ul style="list-style-type: none">Arquitecturas de referencia<ul style="list-style-type: none">Arquitectura de referencia para aplicaciones móvilesArquitectura de referencia para aplicaciones webArquitectura de referencia para infraestructuraArquitectura de referencia para seguridadArquitectura referencia para de integración

Ilustración 8. Wiki Implementada con la base teórica de la metodología ⁸

⁸ Gráfico generado por la autora del trabajo de grado

Esta sección se completa con las definiciones realizadas en la sección del [Marco Conceptual](#). En la Wiki se encuentra el contenido completo e integrado.

6.1.1. TOGAF

6.1.1.1. ¿Por qué Togaf 9.1 es la base del marco de trabajo adaptado?

Teniendo como base la investigación realizada por Cabrera & Lopez (2014) donde hace diversas comparaciones entre seis marcos de trabajar de arquitectura empresarial, se selecciona Togaf 9.1 como el marco de trabajo que se adaptará a la metodología que se desarrollará. Adicionalmente Ruparelia (2010) afirma que Togaf es un es modelo espiral que permite afinar los requisitos durante todo el proceso de desarrollo de la arquitectura, garantizando la continua evaluación de los mismos y generando más valor para el cliente, ya que en cada revisión se enriquecen los prototipos, requerimientos, restricciones, capacidades y de esta forma los riesgos de no cumplir las expectativas del cliente se reducen.

Adicionalmente, desde la experiencia profesional este marco de trabajo permite enfocarse en aspectos tecnológicos y estratégicos de los proyectos de desarrollo de *software* y son precisamente dichos proyectos el corazón y la razón de ser de la metodología.

6.1.2. Archimate

6.1.2.1. Generalidades de Archimate 3.0

En esta sección se explicará cómo usar Archimate 3.0 en el modelamiento de las arquitecturas, sin embargo, se aclara que estos diagramas serán la máscara o primer nivel de modelado. Ya que se pretende que a través de este diagrama se pueda acceder a otros más detallados y específicos, el detalle de los procesos se realizará en BPMN 2.0, la documentación del *software* se realizará con historias de usuario y requerimientos, los diagramas de diseño de sistemas de información y tecnología serán desarrollados en UML 2.0.

Teniendo en cuenta las investigaciones realizadas por Qumer Gill (2015) y Šaša & Krisper (2011) se justifica la afirmación anterior, de esta forma se

garantiza el enriquecimiento de los diagramas y la generación de valor de los mismos. La siguiente imagen muestra la interacción de dichos diagramas:

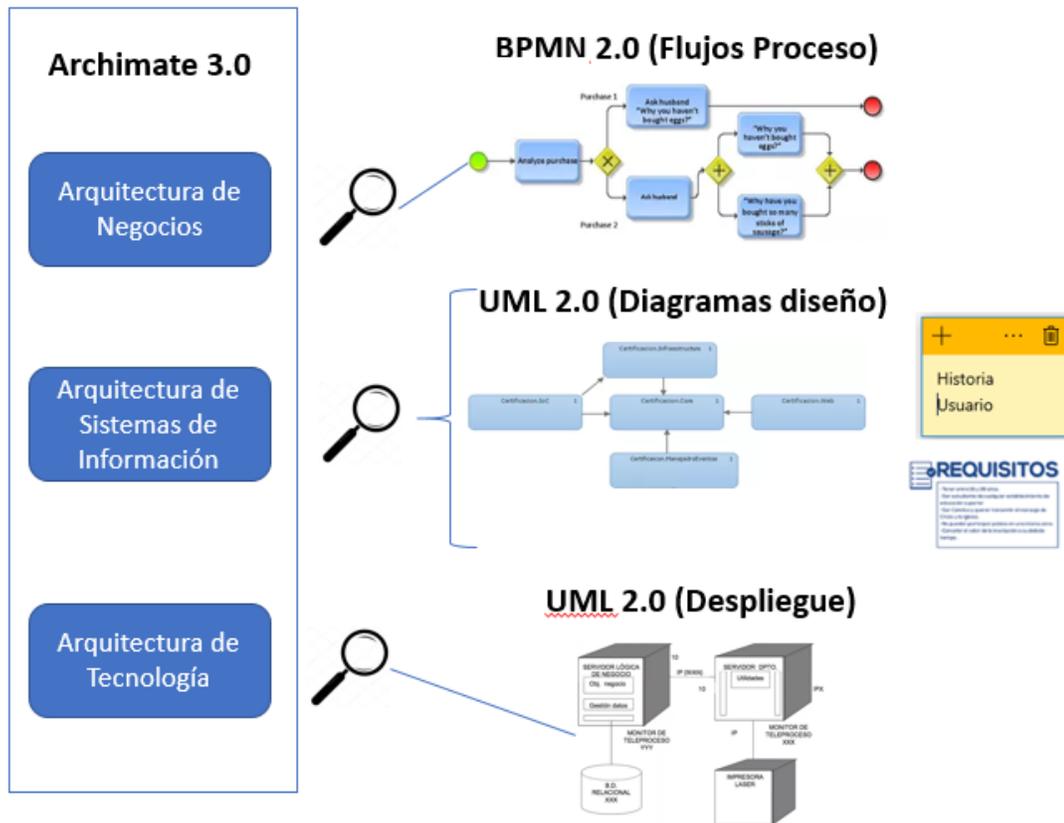


Ilustración 9. Esquema de documentación del marco de trabajo ⁹

Qué es Archimate 3.0

Archimate 3.0 en complemento con los lenguajes de modelado UML 2.0 y BPMN 2.0 cubre todas las fases del ADM de Togaf, como lo muestra la imagen a continuación:

⁹ Gráfico generado por el autor del trabajo de grado

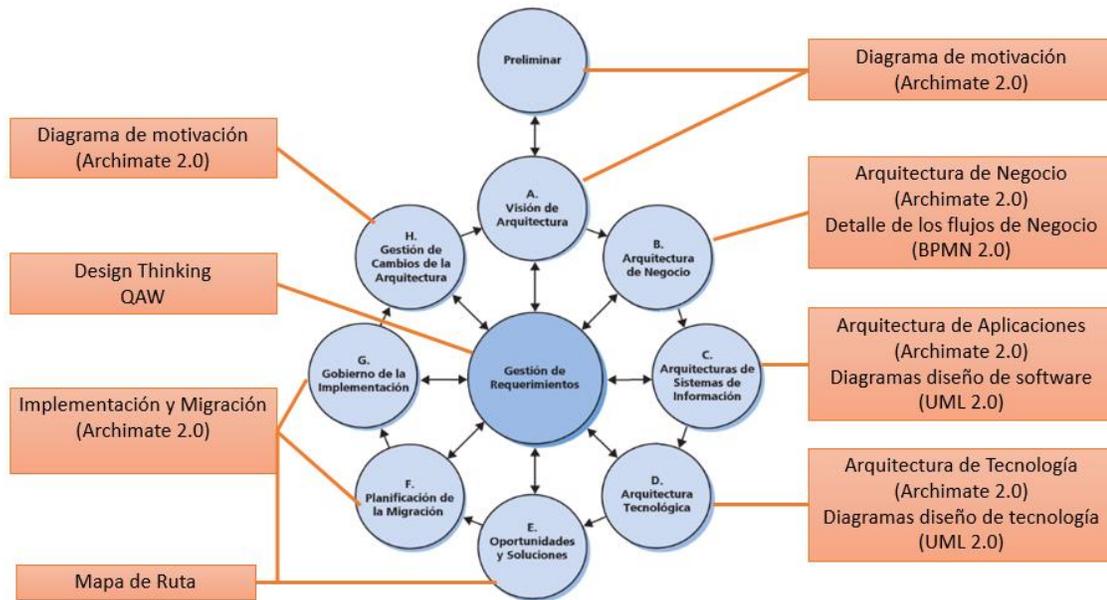


Ilustración 10. Relación entre Togaf 9.1, Archimate 3.0, UML 2.0, BPMN 2.0 y otras técnicas de elicitación de requisitos ¹⁰

¿Para qué se debe usar en la metodología?

Como se mencionó anteriormente Archimate 3.0 es un lenguaje de modelo de arquitectura empresarial, el cual será usado en el marco de trabajo adaptado con el fin de relacionar cada una de las arquitecturas propuestas por Togaf 9.1.

Existen otros lenguajes de modelado de arquitectura empresarial, sin embargo, Archimate 3.0 es desarrollado por *Open Group*, quien también es el desarrollador del marco de trabajo Togaf 9.1. Luego del estudio que se realizó de este lenguaje se determinó que cuenta con todos los componentes para diagramar desde la estrategia hasta las arquitecturas tecnológicas.

6.1.2.2. Conceptos básicos Archimate 3.0

Los conceptos básicos para entender Archimate que se expone en estos documentos tiene como base el conocimiento y la experiencia de los siguientes autores Open Group (2016) y Band, Jonkers, Proper, Quartel, &

¹⁰ Grafico realizado por la autora del trabajo de grado

Lankhorst (2017), sin embargo, para efectos de la documentación del marco de trabajo adaptado la redacción será propia, sencilla, simple y orientado al desarrollo de *software*.

Servicio

Se puede definir el servicio como una unidad funcional que un sistema o proceso expone a su entorno, este puede ser interno o externo. De forma interna se presenta cuando los procesos al interior de una compañía usan este servicio para complementar sus funciones o externo cuando las personas que no hacen parte de la compañía acceden a estos servicios



Ilustración 11. Para este ejemplo el servicio que se presta son las compra por internet ¹¹

Interface

La interface es el medio que permite a los clientes y usuarios acceder al servicio, sin necesidad de conocer la estructura interna de un proceso o una compañía y solamente recibir el beneficio que este genera. Se resalta que la palabra interface es muy común en los sistemas de información cuando dos o más aplicaciones requieren interactuar, sin embargo, para Archimate 3.0 esta definición incluye la capa de negocio.

¹¹ Imagen generada por la autora del trabajo de grado



Ilustración 12. Para acceder a un servicio de compras por internet es necesario contar con un PC, Smartphone o Tablet (Estas son las interfaces) ¹²

Función

Una función es una agrupación de actividades que se ejecutan a raíz de las habilidades y conocimientos que tiene una persona o en caso que sea un sistema de información los eventos que tenga programados.



Ilustración 13 Funciones que debe realizar el sistema de información para realizar la venta ¹³

¹² Imagen generada por la autora del trabajo de grado

¹³ Imagen generada por la autora del trabajo de grado

Proceso

Un proceso describe el comportamiento secuencial o flujo de actividades que se requiere para producir un conjunto de productos y servicios, en el ejemplo que se viene exponiendo un proceso sería graficado de la siguiente forma:



Ilustración 14. Ejemplo de proceso ¹⁴

Elemento Activo

Resumiendo, la definición que entrega Open Group (2016) los elementos activos son aquellos capaz de lanzar acciones y ejecutar comportamientos como son actores de negocio, componentes de aplicación, interfaces, entre otros. Cada elemento se revisará detalladamente en las capas de diagramación

Elemento de comportamiento

Resumiendo, la definición que entrega Open Group (2016) los elementos de comportamiento son aquellos que representa los aspectos dinámicos de la empresa o unidad de actividad. El comportamiento se activa cuando un elemento activo se dispara. Algunos elementos de comportamiento son: procesos, funciones, eventos y servicios. Cada elemento se revisará detalladamente en las capas de diagramación

Elemento Pasivo

Resumiendo, la definición que entrega Open Group (2016) los elementos pasivos son aquellos que sufren cambios o modificaciones como resultado de

¹⁴ Imagen generada por la autora del trabajo de grado

un comportamiento. Generalmente son objetos de información de datos, pero también entra en el grupo los objetos físicos.

6.1.2.3. Capas de diagramación

Capa de negocio

Band, Jonkers, Proper, Quartel, & Lankhorst (2017) afirman que la capa de negocio diagrama los procesos organizacionales de una compañía, permitiendo la identificación de entradas, salidas y actor empresariales, adicionalmente mediante este diagrama se puede evidenciar como funciona una compañía para generar los productos y servicios para clientes internos y externos.

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa y fácil de adoptar.

Elementos Activos

Actor

Open Group (2016) define un actor de negocio como una entidad que se puede encontrar al interior o en el exterior de la organización ya que pueden representar clientes y socios. Cabe resaltar que un actor no solo representa personas, también puede representar departamentos o unidades de negocio. Los actores empresariales interactúan con otros componentes de la siguiente forma:

- Se pueden asignar a funciones de una organización
- Pueden ser agregados a una ubicación.
- Se puede asignar una o más funciones empresariales

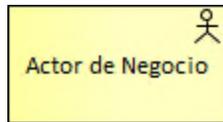


Ilustración 15. Notación de actor de negocio ¹⁵

Rol de Negocio

Open Group (2016) define un rol de negocio como la responsabilidad de realizar un comportamiento específico, los roles empresariales interactúan con otros componentes de la siguiente forma:

- Se pueden asignar a uno o varios actores
- Se puede asignar a uno o varios procesos de negocio
- Una interfaz de negocio o una interfaz de aplicación puede servir a un rol de negocio

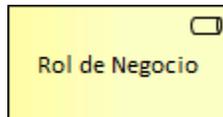


Ilustración 16. Notación de rol de negocio ¹⁶

Colaboración de negocio

Open Group (2016) define una colaboración de negocio como una relación de dos o más roles o actores empresariales que trabajan juntos para realizar un comportamiento colectivo o mostrar la interacción que estos pueden tener. Las colaboraciones empresariales interactúan con otros componentes de la siguiente forma:

- Una interfaz de negocio o una interfaz de aplicación puede servir a una colaboración de negocios
- Una colaboración de negocios puede tener interfaces de negocio

¹⁵ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

¹⁶ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

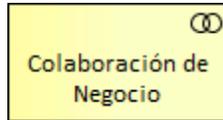


Ilustración 17 Notación de la colaboración de negocio ¹⁷

Interface de Negocio

Open Group (2016) define una la interfaz de negocios como el punto de acceso para que el medio ambiente consuma un servicio de negocio, cabe resaltar que un servicio empresarial puede estar expuesto a través de varias interfaces. Las interfaces de negocio interactúan con otros componentes de la siguiente forma:

- Una interfaz de negocios puede ser parte de un rol o actor de negocio a través de una relación de composición.
- Una interfaz de negocios puede servir a un rol de negocio.
- Una interfaz de negocio puede ser asignada a uno o más servicios empresariales.

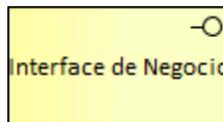


Ilustración 18 Notación de la interface de negocio ¹⁸

Elementos de Comportamiento

Proceso de Negocio

Open Group (2016) define los procesos empresariales como la descripción de un flujo de actividades que se deben ejecutar para producir productos y servicios. Estos son activados por un evento empresarial, proceso empresarial, función de negocio o interacción comercial. Los procesos de negocio interactúan con otros componentes de la siguiente forma:

- Un proceso empresarial puede tener acceso a objetos de negocio.

¹⁷ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

¹⁸ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

- Un proceso empresarial puede realizar uno o más servicios empresariales
- Un proceso empresarial puede utilizar servicios de negocios o servicios de aplicaciones.
- Se puede asignar un rol de empresa a un proceso empresarial

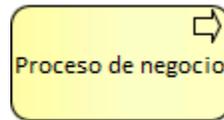


Ilustración 19. Notación del proceso de negocio ¹⁹

Función de negocio

Open Group (2016) define una función de negocio como la representación de las habilidades, competencias y conocimientos específicos de un negocio, la diferencia con los procesos es que estos no tienen un flujo u orden lógico. Describe el comportamiento por un rol de negocio. Las funciones de negocio interactúan con otros componentes de la siguiente forma:

- Una función de negocio puede ser activada por un evento de negocio, proceso de negocio, función de negocio o interacción de negocio.
- Una función de negocio puede tener acceso a objetos de negocio.
- Una función de negocio puede realizar uno o más servicios empresariales y puede ser servida por servicios comerciales, de aplicaciones o de tecnología.
- Se puede asignar un rol de empresa a una función empresarial.

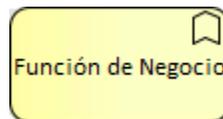


Ilustración 20. Notación de la función de negocio ²⁰

Interacción de negocio

¹⁹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

²⁰ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Open Group (2016) define una interacción comercial como un comportamiento colectivo realizado por una colaboración de dos o más funciones empresariales o múltiples roles. Una interacción comercial puede ser disparada por un evento, proceso de negocio, función de negocio o interacción de negocios. Las interacciones de negocio se relacionan con otros componentes de la siguiente forma:

- Una interacción de negocio puede tener acceso a objetos de negocio.
- Una interacción de negocio puede realizar uno o más servicios empresariales
- Una interacción de negocio puede utilizar servicios de negocios o servicios de aplicaciones.
- Una colaboración de negocio o una colaboración de aplicación pueden ser signadas a una interacción de negocio.

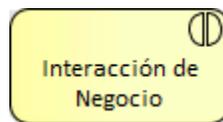


Ilustración 21. Notación de la interacción de negocio ²¹

Evento de negocio

Open Group (2016) define un evento de negocio como el inicio o la interrupción de un proceso o función al interior de una organizacional, cabe resaltar que este evento puede ser interno o externo a la organización. Un evento no tiene duración, es decir, es instantáneo, puede detonarse en un tiempo o momento determinado. Por ejemplo, todos los días a las 8:00 am se generan las conciliaciones con los bancos o cuando se haga una compra electrónica se debe generar la factura de forma automática. Los eventos de negocio interactúan con otros componentes de la siguiente forma:

- Un evento empresarial puede desencadenar por un proceso de negocio, una función de negocio o una interacción de negocio.

²¹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

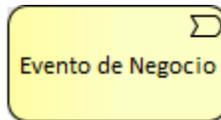


Ilustración 22. Notación de evento de negocio ²²

Servicio de negocio

Open Group (2016) define un servicio de negocios como la exposición de las funcionalidades, procesos o interacciones de negocio ejecutados por los roles y las colaboraciones hacia el entorno externo o interno. Para acceder a un servicio se debe realizar a través de una o varias interfaces. Los servicios de negocio interactúan con otros componentes de la siguiente forma:

- Los servicios de negocio pueden acceder a objetos de negocio.
- Un servicio de negocio se asocia con un valor.
- Un servicio de negocios puede servir a un proceso, función o interacción de negocio.
- Un servicio de negocio puede ser realizado por un proceso, una función o una interacción de negocio.
- Se puede asignar una interfaz de negocios a un servicio de negocio.

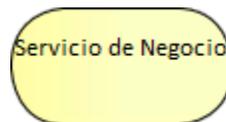


Ilustración 23. Notación de servicio de negocio ²³

Elementos Pasivos

Objeto de Negocio

Open Group (2016) define un objeto de negocio como el encargado de representar un tipo de objeto que pueden tener varias instancias generadas en la ejecución de las operaciones. Puede ser utilizado para representar activos de información del negocio como son plantillas, formatos o documentos, los cuales pueden ser flujos de información de entrada o salida

²² (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

²³ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

de un proceso. Los objetos de negocio interactúan con otros componentes de la siguiente forma:

- Un objeto de negocio puede ser accedido por un proceso, una función, una interacción, un evento o un servicio de negocio.
- Un objeto de negocio puede tener relaciones de asociación, especialización, agregación o composición con otros objetos de negocio.
- Un objeto de negocio puede ser realizado por un objeto de datos.

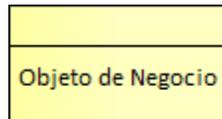


Ilustración 24. Notación de objetivo de negocio ²⁴

Contrato

Open Group (2016) define un contrato como la especificación de los derechos y obligaciones asociados con un producto para la interacción entre un proveedor y un consumidor. Se puede usar para especificar contratos legales, acuerdos de niveles de servicio, acuerdo de funcionalidades, calidad de los servicios. Los contratos interactúan con otros componentes de la siguiente forma:

- Un contrato puede tener una relación de agregación con un producto.

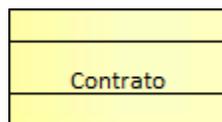


Ilustración 25. Notación de contrato ²⁵

Representación

Open Group (2016) define una representación como un objeto de negocio concreto que transporta información, por ejemplo: mensajes, documentos,

²⁴ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

²⁵ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

correos, entre otros. Un solo objeto de negocio puede tener diversas representaciones.

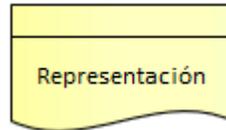


Ilustración 26. Notación de representación ²⁶

Producto

Open Group (2016) define un producto como la generación de los productos o servicios que ofrece una compañía ya sean internos o externos, puede estar acompañado de un contrato que especifique un conjunto de acuerdos para su materialización. Los productos interactúan con otros componentes de la siguiente forma:

- Un producto puede agregar o componer servicios de negocio, servicios de aplicaciones, servicios tecnológicos, objetos de negocio, objetos de datos, objetos tecnológicos y contrato.
- un producto puede agrupar o componer elementos de otras capas.
- Un valor puede estar asociado con un producto.

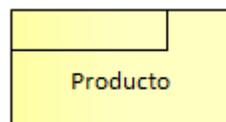


Ilustración 27. Notación de producto ²⁷

Metamodelo de la capa de negocios

The Open Group creó un metamodelo para la capa de negocios con el fin de ayudar a los arquitectos a la hora de genera los diagramas. El objetivo principal del metamodelo es poder ver cada uno de los elementos que hacen parte de la capa de negocios, incluyendo las posibles relaciones que existen entre los componentes, ya sean de esta capa u otras capas.

²⁶ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

²⁷ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

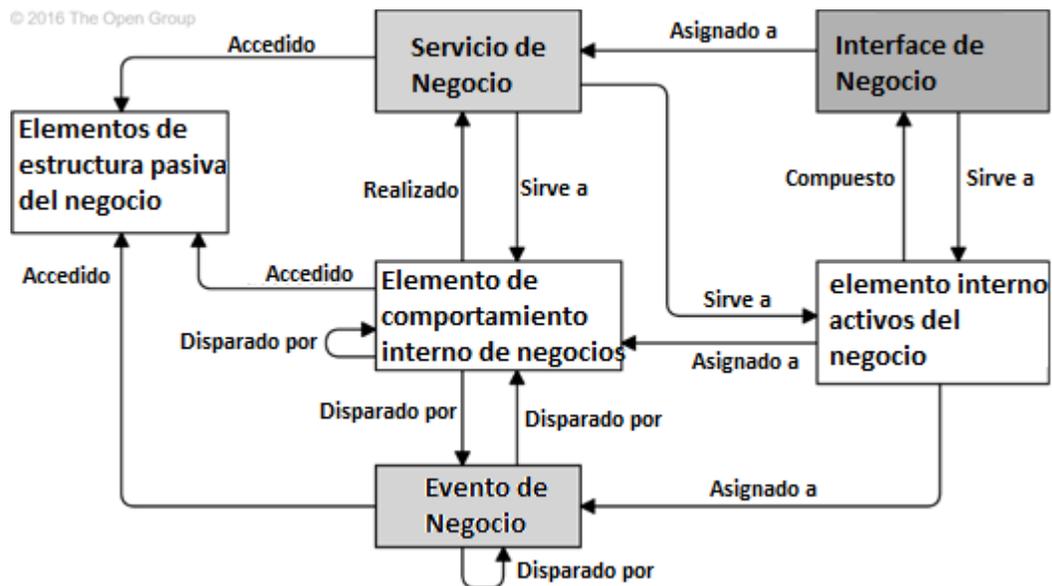


Ilustración 28. Metamodelo de la capa de negocios²⁸

Capa de aplicación

Duarte (2016) afirma que el lenguaje de modelado identifica cada uno de los componentes de *software* que intervienen en el desarrollo de la solución, lo que ayuda a los arquitectos a entender la relación con otros sistemas de información, procesos y tecnología, adicionalmente generar inventarios de cada una de las capas de la arquitectura. La capa de aplicaciones busca evidenciar y graficar cómo los sistemas de información soportan los procesos de negocio, generando trazabilidad e identificando el valor de cada uno al interior de la compañía.

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa fácil de adoptar.

²⁸ Tomado de (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Elementos Activos

Componente de Aplicación

Open Group (2016) define un componente de aplicación como la representación de una aplicación completa o una funcionalidad de la misma, encapsula su comportamiento y datos, expone servicios de aplicación para ser consumidos a través de interfaces de aplicación. Se muestra como una unidad independiente que puede ser desplegable, reutilizable y reemplazable si el negocio así lo requiere. Los componentes de aplicación interactúan con otros componentes de la siguiente forma:

- Un componente de aplicación puede ser asignado una o más funciones de aplicación.
- Un componente de aplicación tiene una o más interfaces de aplicación para exponer sus funcionalidades.
- Un componente de aplicación se puede asignar a una o más funciones de aplicación.
- Las interfaces de aplicación pueden ser usados por un componente de aplicación.

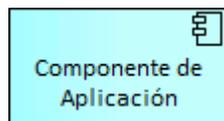


Ilustración 29. Notación de componente de aplicación ²⁹

Colaboración de Aplicación

Open Group (2016) Define una colaboración de aplicación como una representación de dos o más componentes de aplicación que trabajan juntos para llevar a cabo una o varias tareas. En el diagrama se deben especificar cuáles son los componentes que cooperarán para garantizar que la actividad sea realizada. Las colaboraciones de aplicación interactúan con otros componentes de la siguiente forma:

²⁹ (Open Group, 2016)

- La colaboración de aplicaciones es una especialización de dos o más aplicaciones
- Una colaboración de aplicación se puede asignar a una o más interacciones de aplicaciones, interacciones de negocios.
- Una interfaz de aplicación puede exponer una colaboración de aplicación
- Una colaboración de aplicación puede estar compuesta de una o varias interfaces de aplicación.

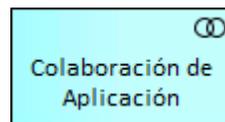


Ilustración 30. Notación de colaboración de aplicación ³⁰

Interface de Aplicación

Open Group (2016) define una interface de aplicación como un punto de acceso a los servicios de aplicación para los usuarios que hacen uso de esta o para el entorno. Una interface de aplicación debe proporcionar un contrato donde se especifican las reglas para su consumo, estas reglas pueden ser parámetros, protocolos, prerrequisito, formatos de datos, entre otros. Las interfaces de aplicación interactúan con otros componentes de la siguiente forma:

- Una interfaz de aplicación puede ser parte de un componente de aplicación a través de la composición.
- Se puede asignar una interfaz de aplicación a los servicios de aplicación.

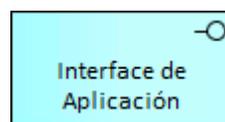


Ilustración 31. Notación de interface de aplicación ³¹

³⁰ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

³¹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Elementos de Comportamiento

Función de aplicación

Open Group (2016) lo define como la representación de las Funcionalidades de un sistema de información, se puede publicar mediante un servicio, para ser consumido posteriormente mediante una interface. Cada función de aplicación describe el comportamiento interno que puede tener un componente de aplicación. Las funciones de aplicación interactúan con otros componentes de la siguiente forma:

- Una función de aplicación realiza uno o más servicios de aplicación.
- Una función de aplicación puede acceder a objetos de datos.
- Un componente de aplicación se puede asignar a una función de aplicación.

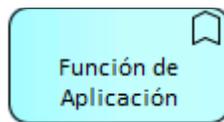


Ilustración 32. Notación de función de aplicación ³²

Interacción de aplicación

Open Group (2016) define una interacción de aplicación como la representación del comportamiento colectivo realizado por una colaboración de dos o más componentes de aplicación. por ejemplo, incluir el patrón de comunicación entre estos componentes. Las interacciones de aplicación interactúan con otros componentes de la siguiente forma:

- Una interacción de aplicación puede tener asignado una colaboración de aplicación.
- Una interacción de aplicación puede realizar un servicio de aplicación.
- Una interacción de aplicaciones puede ofrecer servicios de aplicaciones y servicios de tecnología.
- Una interacción de aplicación puede tener acceso a objetos de datos.

³² (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

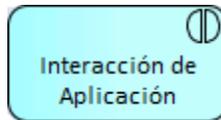


Ilustración 33. Notación de interacción de aplicación ³³

Proceso de aplicación

Open Group (2016) define un proceso de aplicación como la descripción del comportamiento secuencial e interno realizado por un componente de aplicación para entregar un resultado específico. Se conoce por ser una caja negra, ya que su detalle no es conocido por los usuarios finales. Los procesos de aplicación interactúan con otros componentes de la siguiente forma:

- Un proceso de aplicación realiza un servicio de aplicación.
- Un proceso de aplicación puede usar un servicio de aplicación
- Un proceso de aplicación puede acceder a objetos de datos.
- Un proceso de aplicación es realizado por un componente de aplicación.

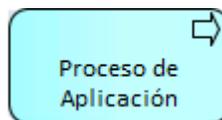


Ilustración 34. Notación de proceso de aplicación ³⁴

Evento de aplicación

Open Group (2016) lo define como un evento que puede activar o interrumpir funciones de aplicación o comportamiento de la misma, este puede ocurrir al interior o exterior de la organización. A diferencia de procesos, funciones e interacciones, un evento es instantáneo; No tiene duración, puede detonarse en un tiempo o momento determinado. Los eventos de aplicación interactúan con otros componentes de la siguiente forma:

- Un evento de aplicación puede desencadenar o ser activado por una función de aplicación, un proceso o una interacción.
- Un evento de aplicación puede tener acceso a un objeto de datos.

³³ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

³⁴ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

- Un evento de aplicación puede estar compuesto de otros eventos de aplicación.

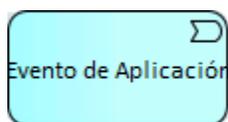


Ilustración 35. Notación de evento de aplicación ³⁵

Servicio de aplicación

Open Group (2016) define un servicio de aplicación como la representación de los servicios que ofrece un sistema de información para ser consumidos por otras aplicaciones, procesos de negocio, funciones de negocio, entre otros, mediante una interface de aplicación que es expuesta al entorno. Los servicios de aplicación interactúan con otros componentes de la siguiente forma:

- Un servicio de aplicaciones puede ser usado por procesos de negocios, funciones empresariales, interacciones comerciales o funciones de aplicaciones.
- Un servicio de aplicación es realizado por una función de aplicación.
- Se puede asignar una interfaz de aplicación a un servicio de aplicación.
- Un servicio de aplicación puede acceder a objetos de datos.

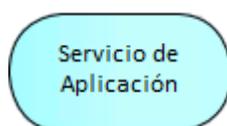


Ilustración 36. Notación de servicio de aplicación ³⁶

Elementos Pasivos

Objeto de datos

Open Group (2016) define el objeto de datos un como la relación directa con las necesidades del negocio, ya que sí el negocio requiere almacenar estructuras de información específicas, la aplicación lo debe de soportar. La

³⁵ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

³⁶ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

estructura de los objetos de datos está directamente relacionada con los objetos de negocios. Ejemplos de objetos de datos son un registro de cliente, una base de datos de clientes, o de una reclamación de seguro. Los objetos de datos interactúan con otros componentes de la siguiente forma:

- Una función de aplicación o proceso puede operar un objeto de datos.
- Un objeto de datos puede ser accedido por una función de aplicación, una interacción de aplicación o un servicio de aplicación.
- Un objeto de datos realiza (materializa) un objeto de negocio.
- Un objeto de datos puede tener relaciones de asociación, especialización, agregación o composición con otros objetos de datos.

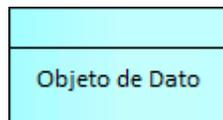


Ilustración 37. Notación de objeto de datos ³⁷

Metamodelo de la capa de Aplicación

The Open Group creó un metamodelo para la capa de aplicación con el fin de ayudar a los arquitectos a la hora de genera los diagramas. El objetivo principal del metamodelo es poder ver cada uno de los elementos que hacen parte de la capa de aplicación, incluyendo las posibles relaciones que existen entre los componentes, ya sean de esta capa u otras capas.

³⁷ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

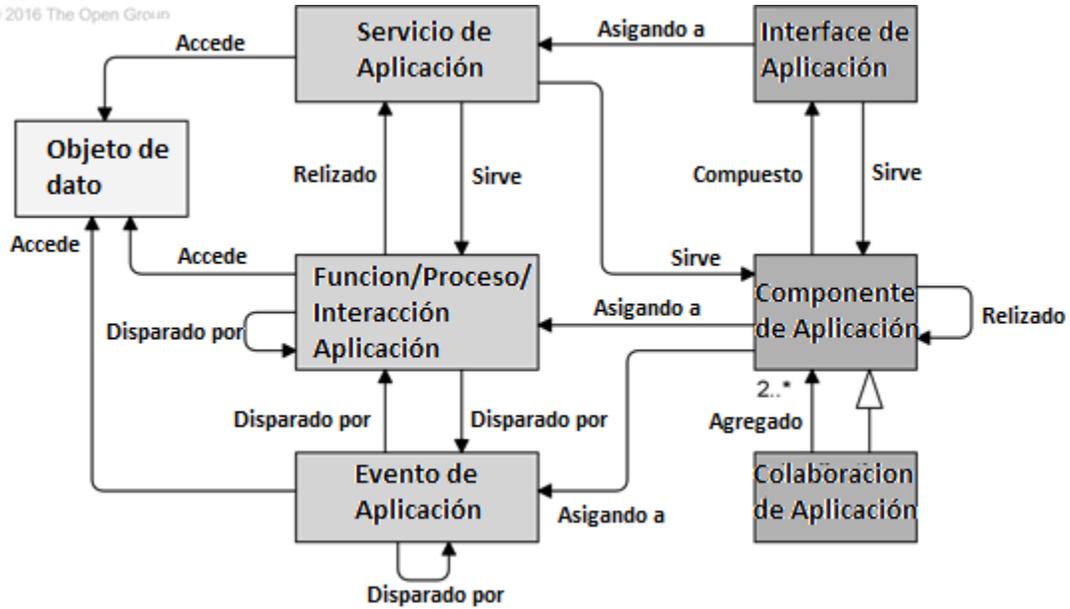


Ilustración 38. Metamodelo de la capa de aplicaciones³⁸

Capa de tecnología

Open Group (2016) resalta que la capa tecnológica se utiliza para modelar la arquitectura tecnológica de la empresa, permite identificar la estructura y la interacción de los servicios de la plataforma y los componentes de la tecnología lógica y física, es decir, contiene toda la infraestructura necesaria para ejecutar sistemas de información, comunicaciones y *software*.

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa y fácil de adoptar.

Elementos Activos

Nodo

Open Group (2016) define un nodo como un recurso de computo sobre el cual los artefactos pueden ser almacenados o desplegados para su ejecución. Un nodo es a menudo una combinación de un dispositivo de *hardware* y *software*

³⁸ Tomado de (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

del sistema, proporcionando así un entorno de ejecución completa. Ejecutan, almacenan y procesan objetos tecnológicos tales como artefactos. Los nodos interactúan con otros componentes de la siguiente forma:

- Los nodos pueden estar interconectados por Path.
- Un nodo puede ser asignado a un artefacto para modelar que el artefacto se despliega en el nodo.
- Los artefactos desplegados en un nodo pueden dibujarse dentro del nodo o conectarse a él con una relación de asignación.



Ilustración 39. Notación de nodo ³⁹

Dispositivo

Open Group (2016) define un dispositivo como un recurso de *hardware* sobre el cual los sistemas de información pueden ser almacenados o desplegados para su ejecución. Un dispositivo es una especialización de un nodo. Se suele utilizar para modelar sistemas de *hardware* como mainframes, computadores o enrutadores. Los dispositivos interactúan con otros componentes de la siguiente forma:

- Los dispositivos pueden ser interconectados por redes.
- Los dispositivos se pueden asignar a artefactos y al *software* del sistema.
- Un nodo puede contener uno o más dispositivos.



Ilustración 40. Notación de dispositivo ⁴⁰

³⁹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁴⁰ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Software del sistema

Open Group (2016) afirma que un *Software* del sistema se encuentra instalado en los nodos o dispositivos. Por ejemplo: sistema operativo, servidor de aplicaciones, servidor de correos, plataformas, entre otros. Se diferencia del *software* de aplicaciones ya que el *software* de infraestructura es la base para que las aplicaciones funcionen. El *software* del sistema interactúa con otros componentes de la siguiente forma:

- Un *software* de dispositivo o sistema se puede asignar a otro *software* del sistema.
- El *software* del sistema se puede asignar a artefactos, para modelar que estos artefactos se implementan en ese *software*.
- Un nodo puede estar compuesto de *software* del sistema.

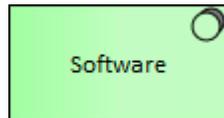


Ilustración 41. Notación del *software* del sistema ⁴¹

Colaboración tecnológica

Open Group (2016) define una colaboración tecnológica como una representación de dos o más nodos que trabajan juntos para llevar a cabo una o varias tareas. En el diagrama se deben especificar cuáles son los componentes que cooperarán para garantizar que la actividad sea realizada. Las colaboraciones tecnológicas interactúan con otros componentes de la siguiente forma:

- La colaboración tecnológica es una especialización del nodo.
- Una colaboración tecnológica puede asignarse a una o más interacciones tecnológicas.
- Una interfaz de tecnología puede ser usado por una colaboración tecnológica, y una colaboración tecnológica puede estar compuesta de interfaces de tecnología.

⁴¹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

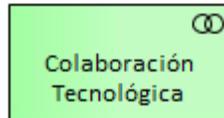


Ilustración 42. Notación de colaboración tecnológica ⁴²

Interface tecnológica

Open Group (2016) define una interfaz de infraestructura como un punto de acceso donde los servicios de infraestructura ofrecidos por un nodo pueden acceder a otros nodos y componentes de la aplicación. Especifica un tipo de contrato que un componente debe cumplir para acceder a este. Por ejemplo: parámetros, protocolos utilizados, precondiciones, pos condiciones, formatos de datos entre otros. Las interfaces tecnológicas interactúan con otros componentes de la siguiente forma:

- Una interfaz de tecnología puede estar compuesta por uno o varios nodos
- Una interfaz de tecnología puede ser asignada a un servicio tecnológico, para exponer ese servicio al medio ambiente.

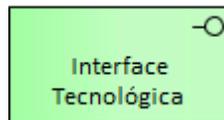


Ilustración 43. Notación de interface tecnológica ⁴³

Trayecto de comunicación (*Path*)

Open Group (2016) define un trayecto de comunicación como una relación entre dos o más nodos, a través del cual estos nodos pueden intercambiar datos. Un trayecto de comunicación se utiliza para modelar las relaciones lógicas entre nodos de comunicación. Se realiza con una o más redes, que representan los enlaces de comunicación físicos. El trayecto de comunicación interactúa con otros componentes de la siguiente forma:

- Se utiliza un trayecto de comunicación para modelar las relaciones lógicas de comunicación entre nodos.
- Un trayecto de comunicación conecta dos o más nodos.
- Un camino es realizado por una o más redes.

⁴² (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁴³ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

- Un trayecto de comunicación puede agregar nodos.



Ilustración 44. Notación de *Path* ⁴⁴

Redes (Network)

Open Group (2016) define una red como una comunicación entre dos o más dispositivos o nodos. Representa la infraestructura de comunicación física por ejemplo LAN, Internet, Intranet. Cabe resaltar que los dispositivos de red como los router, switch, entre otros están inmersos en esta figura. Las redes interactúan con otros componentes de la siguiente forma:

- Se utiliza una red para modelar las relaciones lógicas de comunicación entre nodos.
- Una red conecta dos o más nodos.
- Una red es realizada por una o más redes.
- Una red puede agregar nodos.



Ilustración 45. Notación de redes ⁴⁵

Elementos de Comportamiento

Función de tecnología

Open Group (2016) define una función de infraestructura como una actividad puntual de un nodo, estas son expuestas usando servicios de infraestructura, los cuales son consumidos mediante interfaces de Infraestructura. Por

⁴⁴ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁴⁵ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

ejemplo, un motor de base de datos tiene las siguientes funciones: proporcionar acceso a datos, gestión de datos, entre otros. Adicionalmente describe el comportamiento interno de un nodo. Las funciones de tecnología interactúan con otros componentes de la siguiente forma:

- Una función de tecnología realizar servicios de tecnología.
- Los servicios tecnológicos de otras funciones tecnológicas pueden ser usados por otras funciones tecnológicas.
- Una función de tecnología puede acceder a objetos tecnológicos.
- Un nodo puede ser asignado a una función de tecnología.

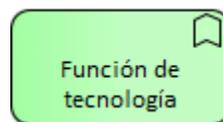


Ilustración 46. Notación de función tecnológica ⁴⁶

Proceso de tecnología

Open Group (2016) define un proceso tecnológico como la representación de una secuencia de comportamientos tecnológicos que ejecuta un nodo para lograr un resultado específico. Se expone usando un servicio de tecnología. Los objetos de tecnología hacen parte del flujo, cumpliendo la función de objetos de entrada o salida. Los procesos de tecnología interactúan con otros componentes de la siguiente forma:

- Un proceso tecnológico realiza servicios tecnológicos.
- Otros servicios tecnológicos pueden ser usados un proceso tecnológico.
- Un proceso tecnológico puede tener acceso a objetos tecnológicos.
- Un nodo puede ser asignado a un proceso de tecnología, lo que significa que este nodo realiza el proceso.

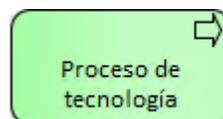


Ilustración 47. Notación de proceso tecnológica ⁴⁷

⁴⁶ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁴⁷ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Interacción de tecnología

Open Group (2016) define una interacción tecnológica como la representación del comportamiento colectivo realizado por una colaboración de dos o más nodos. por ejemplo, incluir el patrón de comunicación entre estos componentes. Las interacciones de tecnología interactúan con otros componentes de la siguiente forma:

- Se puede asignar una colaboración tecnológica a una interacción tecnológica.
- Una interacción de tecnología realiza un servicio tecnológico.
- Los servicios tecnológicos pueden ser usados por una interacción tecnológica.
- Una interacción tecnológica puede tener acceso a artefactos.

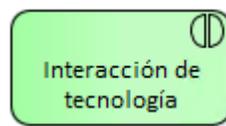


Ilustración 48. Notación de interacción tecnológica ⁴⁸

Evento de tecnología

Open Group (2016) define un evento tecnológico como la activación o interrupción de funciones de aplicación o comportamiento en la infraestructura, este puede ocurrir al interior o exterior de la organización, adicionalmente cabe resaltar que las funciones de tecnología pueden generar eventos que activan otro comportamiento de infraestructura. A diferencia de procesos, funciones e interacciones, un evento es instantáneo, no tiene duración y puede detonarse en un tiempo o momento determinado. Los eventos de tecnología interactúan con otros componentes de la siguiente forma:

- Un evento de tecnología puede desencadenar o ser activado por una función de tecnología, proceso o interacción.
- Un evento de tecnología puede tener acceso a un objeto de datos.

⁴⁸ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

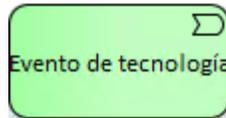


Ilustración 49. Notación de evento tecnológico ⁴⁹

Servicio de tecnología

Open Group (2016) define un servicio de tecnología como la representación de los servicios expuesto al entorno por la infraestructura y las plataformas almacenadas en los servidores, como son: Servicios de Correo, Servicios de Proxy, mensajería, almacenamiento, entre otros. Los nodos y dispositivos pueden acceder a estos servicios mediante interfaces de infraestructura. Puede requerir, usar y producir artefactos. Los servicios de tecnología interactúan con otros componentes de la siguiente forma:

- Un servicio de tecnología puede ser usado por componentes o nodos de la aplicación.
- Un servicio tecnológico es realizado mediante una función o proceso tecnológico.
- Un servicio tecnológico está expuesto por una interface de tecnología.
- Un servicio tecnológico puede acceder a artefactos.

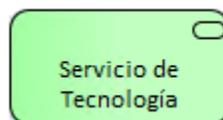


Ilustración 50. Notación de servicio tecnológico ⁵⁰

Elementos Pasivos

Artefacto

Open Group (2016) define un artefacto como la representa un elemento concreto en el mundo físico. Normalmente se utiliza para modelar (*software*) productos tales como archivos de código fuente, ejecutables, scripts, tablas de la base, mensajes, documentos, especificaciones, base de datos, archivos de modelo, entre otros. Un artefacto podría ser utilizado para representar un

⁴⁹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁵⁰ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

componente de datos físico que da cuenta de un objeto de datos. Los artefactos interactúan con otros componentes de la siguiente forma:

- Un componente de aplicación o *software* de sistema es realizado por uno o más artefactos.
- Un objeto de datos se realiza mediante uno o más artefactos.
- Un nodo puede ser asignado a un artefacto para modelar que el artefacto se despliega en el nodo.



Ilustración 51. Notación de artefacto tecnológico ⁵¹

Metamodelo de la capa de Tecnología

The Open Group creó un metamodelo para la capa de tecnología con el fin de ayudar a los arquitectos a la hora de genera los diagramas. El objetivo principal del metamodelo es poder ver cada uno de los elementos que hacen parte de la capa de tecnología, incluyendo las posibles relaciones que existen entre los componentes, ya sean de esta capa u otras capas.

⁵¹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

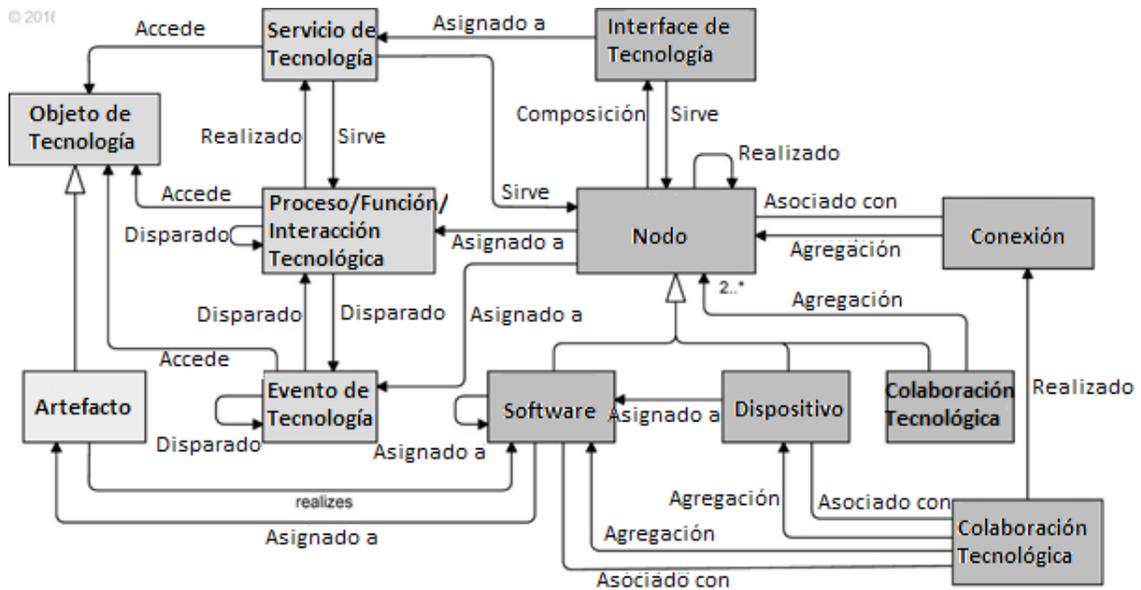


Ilustración 52. Metamodelo de la capa de tecnología⁵²

Motivación

En la investigación realizada por Azevedo, Almeida, Sinderen, Quartel, & Guizzardi (2011) y Duarte (2016) definen la extensión de motivación como componentes adicionales que permiten modelar los objetivos, principios, requisitos y limitaciones que conducen al diseño o el cambio de una arquitectura empresarial. Permite visualizar las motivaciones como impulsores para lograr cambios al interior de una compañía

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa y fácil de adoptar.

⁵² (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Elementos de motivación

Interesado

Open Group (2016) define un interesado como la representación de un individuo, equipo u organización que se beneficia de uno u otra forma del proyecto o arquitectura desarrollada. Sus intereses y preocupaciones deben ser conocidas y gestionadas, al igual que su nivel de poder e influencia.

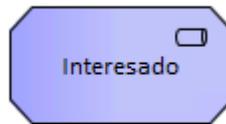


Ilustración 53. Notación de interesado ⁵³

Conductor (*Driver*)

Open Group (2016) define un conductor como la representación de una condición externa o interna que motiva a una organización a definir sus objetivos e implementar los cambios necesarios para alcanzarlos. Los conductores pueden nacer de preocupaciones de los interesados, cambios en factores externos como son económicos, de ley, entre otros, o cambios que la organización decide realizar.



Ilustración 54. Notación de conductor ⁵⁴

Evaluación (*Assessment*)

Open Group (2016) define una evaluación como la representación del resultado de un análisis de la situación de la empresa con respecto a algún conductor. Como resultado se revelan fortalezas, debilidades, oportunidades o amenazas para algún área de interés. Estos deben ser abordados mediante el

⁵³ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁵⁴ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

ajuste de los objetivos existentes o la creación de nuevos, que pueden provocar cambios en la arquitectura de la empresa.

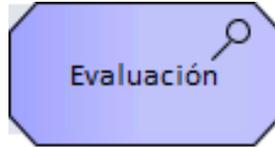


Ilustración 55. Notación de evaluación ⁵⁵

Objetivo

Open Group (2016) define un objetivo como la representación de las intenciones, dirección o estado final deseado para una organización y sus partes interesadas, puede representar cualquier cosa que un interesado pueda desear. Ejemplos de objetivos son: aumentar los beneficios, reducir los costos, aumentar las ventas, entre otros. Los objetivos son usados para medir el éxito de una organización.

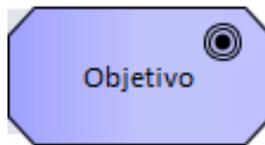


Ilustración 56. Notación de objetivo ⁵⁶

Restricción

Open Group (2016) define una restricción como una representación de un factor que impide u obstaculiza la realización de objetivos. Esta se puede reflejar en la implementación del sistema, implantación del proceso o una restricción en el funcionamiento del sistema.

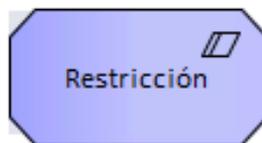


Ilustración 57. Notación de restricción ⁵⁷

⁵⁵ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁵⁶ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁵⁷ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Principio

Open Group (2016) define un principio como una fuerte relación con las metas y los requisitos. Los principios definen las propiedades de los sistemas, son amplios en su alcance y más abstractos que los requisitos. Un principio define una propiedad general que se aplica a cualquier sistema en un determinado contexto y es motivado por algún objetivo o conductor.

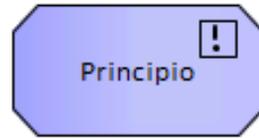


Ilustración 58. Notación de principio ⁵⁸

Valor

Open Group (2016) define un valor como la ganancia que se obtiene al vender o poner a disposición algún producto o servicio, se expresa a menudo en términos de dinero, sin embargo, cabe resaltar que con el tiempo se ha reconocido que el valor no monetario también es esencial para las empresas.



Ilustración 59. Notación de valor ⁵⁹

Metamodelo de motivación

The Open Group creó un metamodelo para la capa de motivación con el fin de ayudar a los arquitectos y analistas de negocio a la hora de genera los diagramas. El objetivo principal del metamodelo es poder ver cada uno de los elementos que hacen parte de la capa de motivación, incluyendo las posibles relaciones que existen entre los componentes, ya sean de esta capa u otras capas.

⁵⁸ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁵⁹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

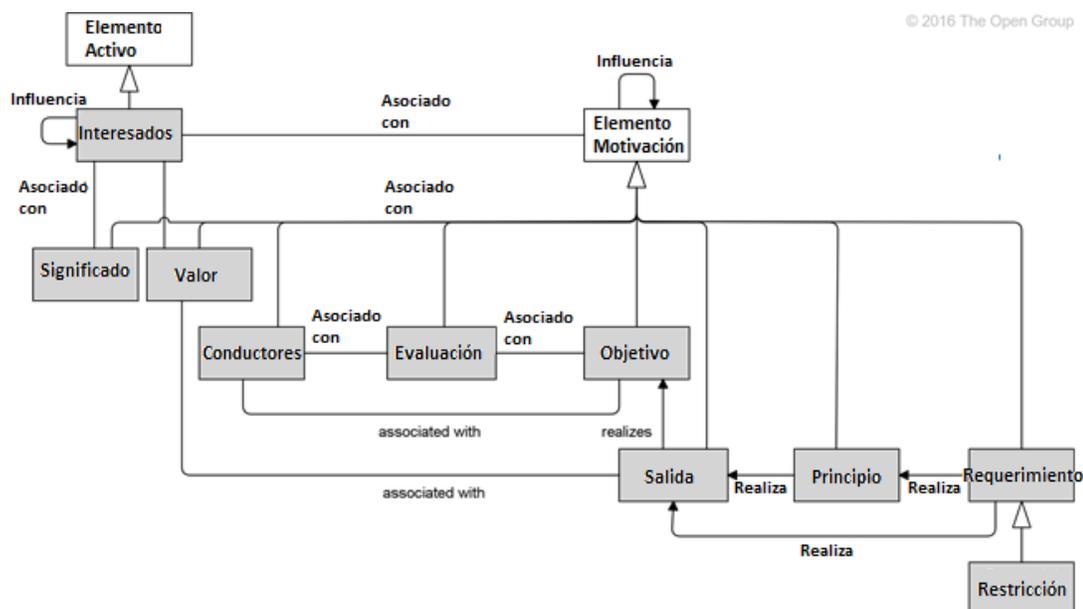


Ilustración 60. Metamodelo de la capa de motivación⁶⁰

Implementación y migración

Partiendo de la afirmación de Duarte (2016) y Jonkers, Berg, Iacob, & Quartel (2010) se concluye que la capa de implementación y migración son componentes que permite modelar transición de una arquitectura a otra, permiten visualizar que elementos serán eliminados, cuales ingresarán al modelo y/o continuarán en el mismo. Adicional a lo anterior permite visualizar las brechas y los momentos de estabilidad que se tienen en la transición.

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa y fácil de entender.

⁶⁰ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Elementos de implementación y migración

Paquete de trabajo

Open Group (2016) define un paquete de trabajo como un conjunto de tareas que tienen claramente definidas fecha de inicio y fecha de finalización, adicionalmente materializan un conjunto de objetivo o entregables definidos previamente por los interesados.

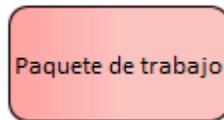


Ilustración 61. Notación de paquete de trabajo ⁶¹

Entregable

Open Group (2016) define una entrega como una representación de un resultado definido de un paquete de trabajo, por ejemplo, informes, documentos, servicios, *software*, productos físicos, cambios organizacionales, implementación de arquitecturas etc.



Ilustración 62. Notación de entregable ⁶²

Meseta

Open Group (2016) define una meseta como una representación de una arquitectura en un momento del tiempo, es decir, en las fases de arquitectura de negocios, aplicaciones y tecnología se tiene una arquitectura actual y un destino; en la fase oportunidades y soluciones se diseñan arquitecturas de transición donde se evidencia cómo se está cumpliendo el objetivo hasta llegar a la arquitectura destino

⁶¹ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

⁶² (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

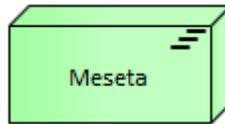


Ilustración 63. Notación de meseta

Brecha

Open Group (2016) define una brecha como una representación de una o varias diferencias entre dos mesetas, entendiéndose por meseta el ASIS y el TOBE, es un insumo importante para la posterior implementación y planificación de la migración.



Ilustración 64. Notación de brecha ⁶³

Metamodelo de implementación y migración

The Open Group creó un metamodelo para la capa de implementación y migración con el fin de ayudar a los arquitectos a la hora de genera los diagramas. El objetivo principal del metamodelo es poder ver cada uno de los elementos que hacen parte de la capa de implementación y migración, incluyendo las posibles relaciones que existen entre los componentes, ya sean de esta capa u otras capas.

⁶³ (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

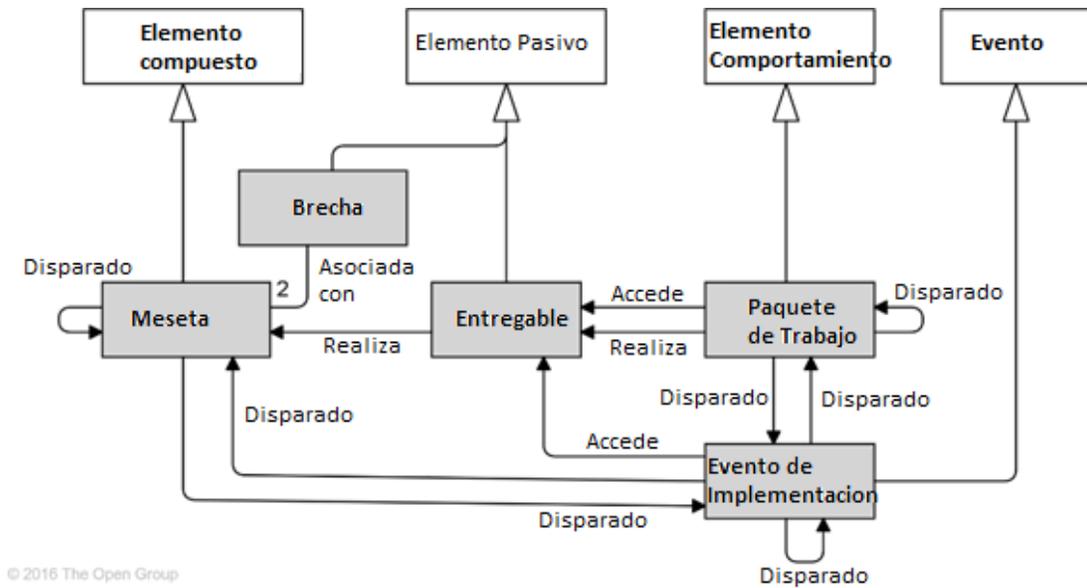


Ilustración 65, Metamodelo de la capa de implementación y migración ⁶⁴

6.1.3. BPMN 2.0

¿Para qué se debe usar en la metodología?

Teniendo en cuenta las investigaciones realizadas por Qumer Gill (2015) y Šaša & Krisper (2011) el lenguaje de modelado BPMN 2.0 Complementa el modelado de la arquitectura de negocio realizada en con Archimate 3.0.

Se pretende que el detalle de los procesos esté modelado en un lenguaje reconocido a nivel mundial y que permita el fácil entendimiento para los arquitectos de procesos y cada uno de los miembros de una organización. Adicionalmente permite enriquecer los procesos con detalles importantes que Archimate no permite modelar.

Componentes en la diagramación de BPMN 2.0

Existe gran cantidad de bibliografía para describir cada uno de los componentes de la notación de BPMN 2.0, en este trabajo se tiene como base las definiciones realizadas por los siguientes autores: Weilkiens, Weiss, &

⁶⁴ Tomado de (Open Group, 2016) – Traducción realizada por la autora del trabajo de grado

Grass (2011), OMG (2011) y Freund, Rücker, & Hitpass (2014), por lo que cada definición será una redacción propia, pero fundamentada en los libros mencionados anteriormente.

En los próximos numerales se explicará cada uno de los objetos que se trabajará en el marco de trabajo adaptado, cabe resaltar que no se tiene contemplado usar todos los componentes ya que el objetivo de esta metodología es que sea ligera, precisa y fácil de entender.

Eventos

Los eventos son representados por un círculo y son considerados ocurrencia, disparadores o sucesos durante la ejecución del proceso, cabe resaltar que esta ocurrencia puede ser esperada o inesperada. Los eventos se dividen en tres tipos, de inicio (son los que inician un flujo), intermedios (hechos que ocurren en la mitad del proceso) y finales (terminan un flujo). Cada tipo de evento tiene diversas clasificaciones que se explicarán a continuación:

Eventos	INICIO			INTERMEDIOS			FIN	
	ALTO NIVEL	EVENTO INTERRUPTOR DE SUBPROCESO	EVENTO NO INTERRUPTOR DE SUBPROCESO	CAPTURA	ADJUNTO INTERRUPTOR	ADJUNTO NO INTERRUPTOR	LANZAMIENTO	
Ninguno: Evento sin tipo, indican el punto de inicio, el estado los cambios o estados finales.								
Mensaje: Recepción y envío de mensajes.								
Temporal: Puntos en el tiempo, lapsos, límites (timeouts). Pueden ser eventos únicos o cíclicos.								
Escalable: Cambio a un nivel más alto de responsabilidad.								
Condicional: Reacción a cambios en las condiciones de negocios o integración de reglas de negocio.								
Enlace: Conectores fuera de página. Dos conectores de enlace equivalen a un flujo de secuencia.								
Error: Captura y lanzamiento de errores conocidos con nombre.								
Cancelación: Reacción a la cancelación de una transacción/ Solicitud de cancelación.								
Compensación: Manejo/Solicitud de compensación.								
Señal: Intercambio de señales entre procesos. Una señal puede ser capturada varias veces.								
Múltiple: Captura uno de un conjunto de eventos. Lanza todos los eventos definidos.								
Paralela Múltiple: Captura todos los eventos de un conjunto de eventos en paralelo.								
Terminación: Terminación inmediata del proceso.								

Ilustración 66. Resumen de eventos en BPMN 2.0 ⁶⁵

Actividades (Tareas y Subtareas)

Las actividades son acciones que los actores deben ejecutar en el proceso, cabe resaltar que sus descripciones deben iniciar con un verbo en infinitivo. Adicionalmente son consideradas la espina dorsal de los procesos. Existen varios tipos de actividades que se explicarán a continuación:

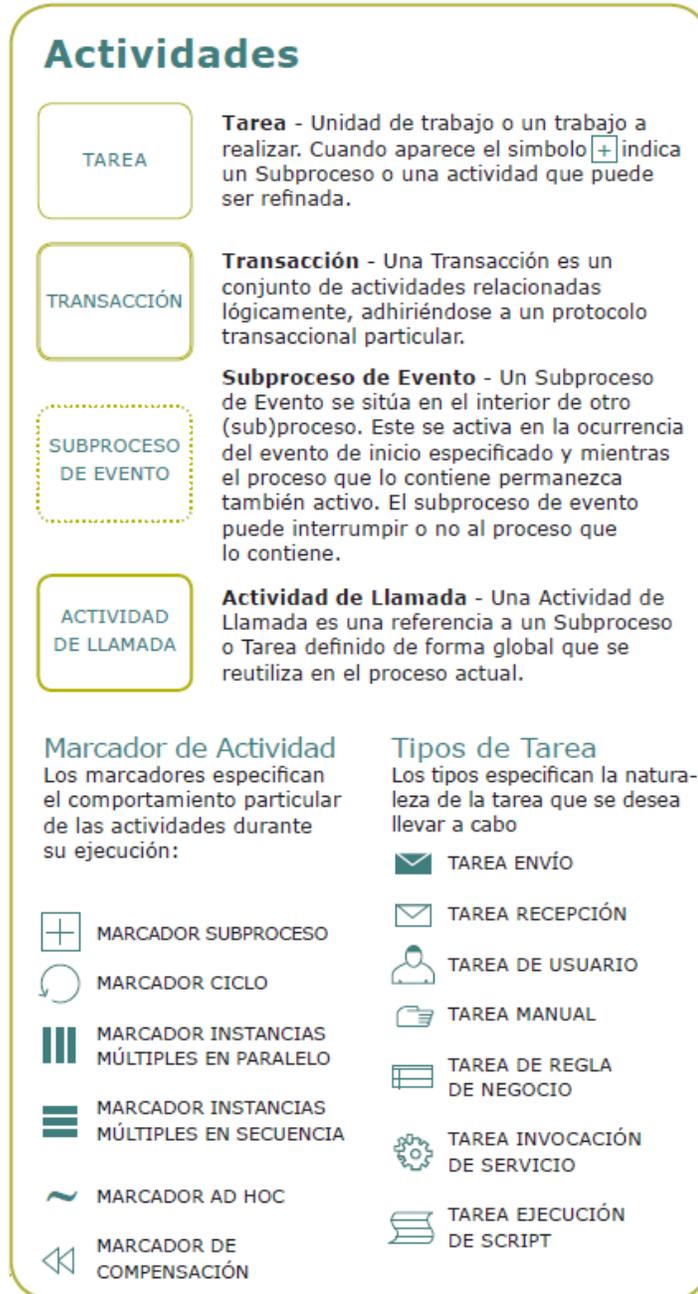


Ilustración 67. Resumen de actividades ⁶⁶

⁶⁵ Tomado de (VisumPoint EnterpriseArchitecture, s.f.)

⁶⁶ Tomado de (VisumPoint EnterpriseArchitecture, s.f.)

Compuertas

Las compuertas son usadas para controlar las convergencia o divergencia de diversos flujos de procesos. Son modeladas para tomar decisiones, bifurcaciones y uniones.

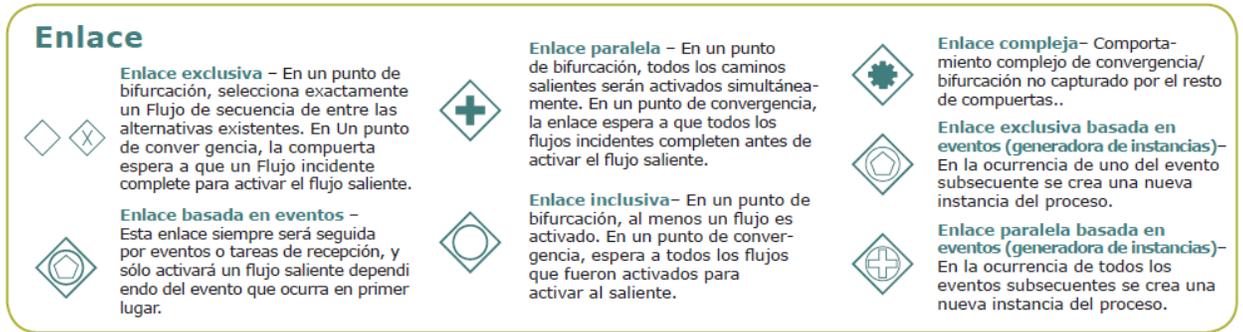


Ilustración 68. Resumen de compuertas ⁶⁷

Objetos de Conexión

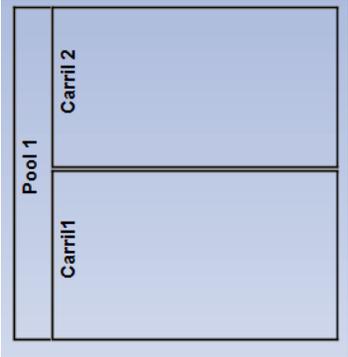
Los objetos de conexión permiten modelar la relación y secuencia existente entre actividades, eventos y compuertas, ya sea que se encuentren ya sea que se encuentren en el mismo o en diferentes pool y flujos de información.

Componente	Descripción	Símbolo
Flujo de secuencia	Permite la conexión entre actividades, eventos y compuertas en un mismo pool	
Flujo de mensaje	Permite la conexión de actividades, eventos y compuertas entre componentes de diferentes pools	
Asociación	Permite asociar componentes a los propios de BPMN 2.0 con los artefactos, incluso modelar flujos de información.	

⁶⁷ Tomado de (VisumPoint EnterpriseArchitecture, s.f.)

Swimlanes

Lo *swimlanes* son usados para clasificar u organizar actividades, el objetivo es diagramar las diferentes responsabilidades de áreas, roles, personas, entre otros.

Componente	Descripción	Símbolo
Pool	Representa un participante o una entidad al interior de un proceso	
<i>Lanes</i>	Los <i>lanes</i> o carriles son una división al interior de un pool, con el fin de organizar o categorizar roles, actividades e incluso personas.	

Artefactos

Son objetos que recrean los documentos o información que hace parte de un proceso, como por ejemplo la entrada o salida de documentos, variables, mensajes, entre otros.



Ilustración 69. Resumen de artefactos ⁶⁸

6.1.4. UML 2.0

¿Para qué se debe usar en la metodología?

Los diagramas UML estarán orientados al modelado de *software* y a la infraestructura tecnológica, estos serán usados para detallar la capa de aplicaciones, datos y tecnología que se diagrama con Archimate 3.0, es decir, serán los encargados de mostrar el detalle y diseño del *software* empresarial y las plataformas tecnológicas.

⁶⁸ Tomado de (VisumPoint EnterpriseArchitecture, s.f.)

Cabe resaltar que no se usarán todos los diagramas que UML ofrece, por lo tanto, en esta guía solo se hará énfasis los modelos que se usarán en la metodología adaptada.

6.1.4.1. Diagrama de Componentes

¿Para qué usarlo en el marco de trabajo?

El diagrama de componentes se usa en el marco de trabajo adaptador para conocer el detalle de la estructura y dependencia de los mismos, permite identificar el alcance del desarrollo, ya que es posible identificar los componentes propios del sistema de información empresarial y aquellos que son de autoría de otros proveedores.

¿Cuándo usarlo?

Los diagramas de componentes serán usados en el marco de trabajo adaptado para describir la relación que tiene un sistema de información empresarial con otros componentes, ya sean externos, de otras aplicaciones e incluso con sus mismos elementos. De esta forma el arquitecto podrá diseñar y exponer mediante un diagrama las relaciones que existen y los medios que usan para garantizar su comunicación.

Adicionalmente los componentes del diagrama de componentes pueden ser usado para crear diagramas separados en capas, y de esta forma los desarrolladores entiendan como deben estructurar los proyectos y el código.

Componentes que conforman el diagrama

Componente

Kimmel (2008) define los componentes como código autónomo capaz de ejecutar una funcionalidad en particular, adicionalmente se pueden desplegar de forma independiente. Muchos de estos componentes se pueden usar mediante las interfaces, por lo tanto, los otros componentes invocan solo la interface y de esta forma se benefician sin necesidad de conocer la implementación.

Interface

Weilkiens, Systems Engineering with SysML/UML (2008) y Rumbaugh, Jacobson, & Grady (2004) Definen una interfaz como la descripción del comportamiento del objeto o clase sin conocer la implementación o estado del mismo. Una o más clases puede realizar una interfaz, es necesario que cada una de las operaciones de la interface sean implementadas por la clase.

Componentes con puertos

Spark System (2015) afirma que el uso de estos componentes especifica una combinación de componentes sencillos e interfaces, donde se puede modelar si un componente está exponiendo un servicio o lo está consumiendo. Cabe resaltar que un componente puede exponer un servicio mediante una interface y al mismo tiempo usar otra interface de otro componente.

Paquete

Kimmel (2008) y Rumbaugh, Jacobson, & Grady (2004) afirman que un paquete tiene la apariencia de una carpeta de archivos, se usa para representar un nivel elevado de clasificación. Puede ser usado como un espacio de nombre o como un mecanismo con un propósito general para organizar elementos en grupos, establecer la propiedad de los elementos y proporcionar nombres únicos para referenciar elementos.

Dependencia

Rumbaugh, Jacobson, & Grady (2004) define una dependencia como una relación entre dos elementos en los cuales un cambio de un elemento puede afectar o suministrar la información necesaria para el otro elemento.

Generalización

Kimmel (2008) define la generalización como una herencia, de esta forma una clase hija puede heredar los atributos y operaciones de su clase padre, cabe resaltar que solo tiene acceso a los miembros públicos o protegidos.

Asociación

Kimmel (2008) define una asociación como la conexión de dos objetos o dos instancias de objetos en un sistema. Rumbaugh, Jacobson, & Grady (2004) afirma que estas relaciones pueden enriquecerse mediante las leyendas para identificar la relación puntual que existe entre los componentes, adicionalmente puede contener la cardinalidad, dirección y restricciones.

Realización

Kimmel (2008) define la realización como la relación entre una clase concreta y una interface, donde la clase concreta realiza o implementa las operaciones definidas en la interface.

6.1.4.2. Diagrama de distribución o despliegue

¿Para qué usarlo en el marco de trabajo?

El diagrama de distribución o despliegue se usa en el marco de trabajo adaptador para conocer dónde se almacenan los componentes físicamente en los servidores de la empresa, este diagrama alimenta la trazabilidad entre los componentes arquitectónicos y en caso de hacer cambios o modificaciones es posible visualizar y tener la trazabilidad de cuáles sistemas de información se podrían ver afectadas.

¿Cuándo usarlo?

Este diagrama se usará en el marco de trabajo adaptado para diagramar dónde están ubicados físicamente los componentes, es decir, en dispositivos, servidores, en la nube, entre otros. Y como se comunican entre ellos, es decir,

conectividad, *networking*, entre otros. Este diagrama se usará para cada ambiente donde el sistema de información se despliegue.

Componentes que conforman el diagrama

Nodo

Kimmel (2008) define un nodo como un cubo que representa un dispositivo físico, el cual se puede interpretar como un computador, servidores físicos, servidores virtuales, entre otros. Al interior de los nodos se almacenan los componentes o servicios que prestados por los nodos.

Componente

Kimmel (2008) define los componentes como código autónomo capaz de ejecutar una funcionalidad en particular, adicionalmente se pueden desplegar de forma independiente. Muchos de estos componentes se pueden usar mediante las interfaces, por lo tanto, los otros componentes invocan solo la interface y de esta forma se benefician sin necesidad de conocer la implementación. Kimmel (2008) afirma que los componentes cuando ya están desplegados también se conocen como artefactos.

Artefacto

Rumbaugh, Jacobson, & Grady (2004) define un artefacto como la implementación de un componente como por ejemplo una página web, base de datos, ejecutables, script, entre otros.

Asociación

Kimmel (2008) define una asociación como la conexión de dos objetos o dos instancias de objetos en un sistema. Rumbaugh, Jacobson, & Grady (2004) afirma que estas relaciones pueden enriquecerse mediante las leyendas para identificar la relación puntual que existe entre los componentes, adicionalmente puede contener la cardinalidad, dirección y restricciones. En el diagrama de despliegue esta relación representa la conectividad que se tiene entre nodos y se conoce como un *communication path*.

Dependencia

Kimme (2008) define una dependencia como la asignación de un artefacto a un destino o despliegue. En el diagrama de distribución o despliegue se conoce como relación de despliegue.

6.1.4.3. Diagrama de estados

¿Para qué usarlo en el marco de trabajo?

El diagrama de estados se usa en el marco de trabajo adaptador para conocer los cambios de estados que puede tener las entidades de negocio según las reglas de la misma. Estos cambios de estado van a estar relacionados directamente con funciones manuales y automáticas ejecutadas en el sistema de información empresarial.

¿Cuándo usarlo?

El diagrama de estados se usará cuando un objeto cambie más de dos veces de estado, el objetivo es que el arquitecto pueda plasmar en un gráfico del flujo de estados y las acciones que disparan su cambio, de esta forma todos los interesados del proyecto tendrán claro los factores que intervienen en cada cambio de estado.

Componentes que conforman el diagrama

Estado

Spark System (2015) afirma que un estado debe ser representado por un cuadro con las puntas redondeadas, donde se describa al estado que cambia el objeto.

Estado inicial o final

Spark System (2015) define el estado inicial y final como los puntos de entrada o salida que están marcados en el flujo, es decir, son las acciones que hacen que un objeto inicie o termine el flujo de la máquina de estados. El inicio se representa como un círculo relleno y el fin con un círculo con un punto en el medio.

Transiciones

Spark System (2015) define la transición como el cambio de un estado a otro, se representa con una flecha con una línea continua. Este símbolo permite al lector del diagrama entender el paso entre estados y la razón de su cambio, es decir, conocer el disparados que lo genera.

6.1.5. SCRUM

6.1.5.1. *Generalidades de scrum*

Qué es scrum

Basado en las definiciones de los siguientes autores Satpathy (2016) y Schwaber & Sutherland (2013) se define Scrum como una metodología ágil que permite centrarse en lo importante para dar paso a un producto de alto valor. Este resultado es el esfuerzo de un equipo comprometido y auto gestionado, donde cada integrante conoce sus responsabilidades, tanto personales como colectivas. Adicionalmente se desarrolla la comunicación efectiva ya que cada miembro del equipo debe conocer el trabajo de sus compañeros y cómo integrarlo al suyo.

La metodología se compone de roles al interior del equipo de desarrollo, eventos y artefactos, los cuales serán detallados en la sección de la metodología ágil. La siguiente imagen es tomada de (Scrum.org, 2005) donde se resume todo el marco de trabajo de Scrum

The Agile: Scrum Framework at a glance

Información de los ejecutivos,
el equipo, los implicados,
los clientes, los usuarios, etc.

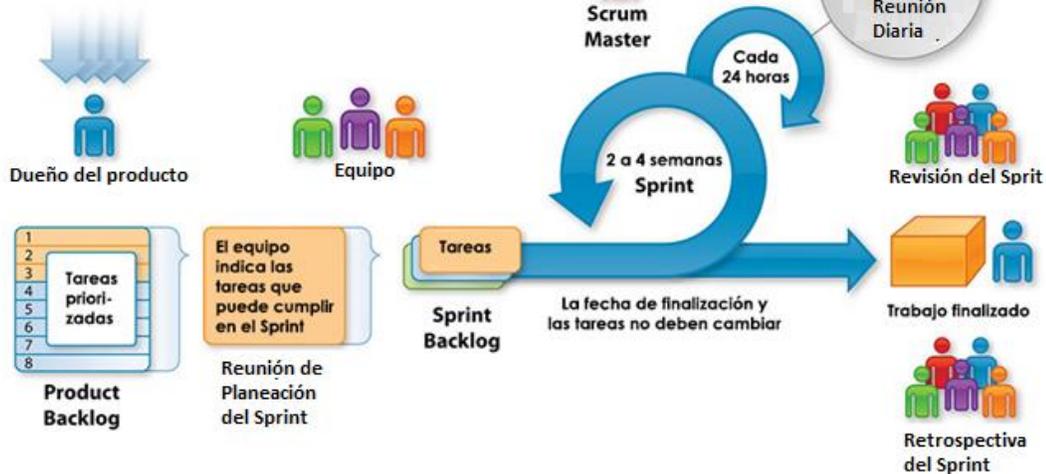


Ilustración 70. Marco de trabajo de Scrum ⁶⁹

¿Para qué se debe usar en la metodología?

El papel que juega Scrum en el marco de trabajo adaptado se materializa en la codificación del proyecto de desarrollo de *software* empresarial, con el fin de tener metas claras en cada sprint y garantizar las entregas funcionales a los usuarios finales.

Se seleccionó Scrum porque es una metodología que ha venido tomando mucha fuerza en el desarrollo de *software* y forma equipos auto gestionados, ya que cada integrante del equipo de desarrollo se apodera de sus entregables desde su inicio hasta su puesta en ejecución.

Se resalta la afirmación realizada por Schwaber & Sutherland (2013) en la visión general donde dice que Scrum es ligero, fácil de aprender y difícil de aplicar, comparándolo con el marco de trabajo adaptado. El marco de trabajo adaptado es ligero en cuanto a la documentación que se tiene en su ejecución, fácil de aprender a pesar que tiene como base una metodología robusta como es Togaf 9.1 y difícil de aplicar, ya que los proyectos se deben de mirar como un todo incluyendo la estrategia, procesos y tecnología de la compañía donde se aplique.

⁶⁹ (Scrum.org, 2005) – Traducción realizada por la autora del trabajo de grado

Teniendo en cuenta las experiencias de Godoy, Belloni, Kotynsky, Dos Santos, & Sosa (2014), Santos, Kroll, Sales, Fernandes, & Wild (2016) y Shahzeydi & Gandomani (2016) es viable apostarle a Scrum por sus resultados exitosos. Los tres primeros casos afirman que Scrum es una metodología adaptable y flexible para desarrollo de *software* o para adaptar a otras metodologías existentes. El segundo y tercer caso hacen énfasis en continuar con pruebas de concepto de la metodología para afinar su uso.

Para terminar, se hace énfasis en la afirmación de Satpathy (2016) donde argumenta porqué se debe usar Scrum:

- Adaptabilidad
- Transparencia
- Retroalimentación continua
- Mejora continua
- Entrega continua de valor
- Ritmo sostenible
- Entrega anticipada de alto valor
- Procesos de desarrollo eficiente
- Motivación
- Resolución de problemas de forma rápida
- Entregables efectivos
- Centrado en el cliente
- Ambiente de alta confianza
- Responsabilidad colectiva
- Ambiente innovador

6.1.5.2. Metodología Scrum

Roles en sprint

Product Owner

Schwaber & Sutherland (2013) define al *Product Owner* o dueño del producto como la persona que conoce el proceso, las reglas de negocio, las necesidades de la empresa y tiene claro cuál es el alcance del proyecto. Por lo anterior es el encargado de definir el *product backlog*, el cual se construirá en la fase de requerimientos del marco de trabajo adaptado.

Debe tener habilidades de comunicación efectiva ya que siempre debe estar disponible para responder las pregunta que surjan. Adicionalmente es la persona que elige los entregables que se van a desarrollar en cada sprint ya que conoce la prioridad del *product backlog*.

Scrum Master

Schwaber & Sutherland (2013) define el *Scrum master* como la persona encargada de garantizar que la metodología se esté ejecutando, es decir, garantizar la completitud del equipo de trabajo, el cumplimiento de los eventos, ayuda a resolver los problemas que se presenten en los sprint y canaliza los problemas externos del proyecto con el fin de que el equipo no se afecte por estos factores.

Equipo de desarrollo

Schwaber & Sutherland (2013) define el equipo de desarrollo como el encargado del diseño y desarrollo del *software* empresarial, se caracteriza por ser un equipo auto gestionado, enfocado en los objetivos y es multidisciplinario, es decir, aunque las personas tengan una especialidad pueden cubrir otros roles que el proyecto esté necesitando. Los retos más grandes del equipo de trabajo son la comunicación efectiva y la consolidación como equipo de desarrollo.

Eventos en sprint

Sprint

Schwaber & Sutherland (2013) define el sprint como un rango de tiempo que incluye todas las ceremonias o reuniones que propone Scrum, pero estas actividades no deben exceder un mes calendario. Inicia con la reunión de planeación, donde se conoce el alcance para el *sprint*, cada día se hacen reuniones cortas de seguimiento, se desarrollan los entregables, se entregan al cliente el incremento del producto completamente funcional y para finalizar se hace una retrospectiva, con el fin de garantizar la mejora continua del equipo y resaltar las lecciones aprendidas.

Planeación de *sprint*

Schwaber & Sutherland (2013) afirma que la ceremonia de planeación consiste en la identificación del alcance que será desarrollado en el *sprint*. Debe estar todo el equipo del proyecto presente ya que cada uno conoce su capacidad de desarrollo y de esta forma comprometerse con el incremento del sistema de información empresarial.

La planeación está compuesta de dos actividades macro, la primera es la creación del *sprint backlog*, que es conocido como una porción del *product backlog* y la segunda es la definición de cómo se van a desarrollar el *sprint backlog*.

Daily Scrum

Schwaber & Sutherland (2013) afirma que la reunión diaria es una actividad de 15 minutos, donde cada integrante del equipo de desarrollo informa que avances hizo el día anterior, con qué se compromete para el día actual y si ha tenido algún inconveniente o problema. Si el problema no es muy extenso se discuten soluciones con el resto del equipo de desarrollo, de esta forma se aprovecha el conocimiento de los compañeros para solucionar problemas o encontrar alternativas.

Revisión del *sprint*

Schwaber & Sutherland (2013) afirma que la revisión del *sprint* es la entrega formal del incremento del sistema de información empresarial a todos los involucrados, incluyendo los desarrolladores, usuarios funcionales, directivos y líderes del proyecto. El objetivo es hacer un pequeño demo o presentación de las nuevas funcionalidades, las cuales están disponibles para su uso inmediato.

Retrospectiva del *sprint*

Schwaber & Sutherland (2013) afirma que la retrospectiva del *sprint* es una reunión del equipo Scrum donde cada integrante hace un aporte de mejora y/o resalta las buenas prácticas ejecutadas en el proyecto, es decir, es un espacio de lecciones aprendidas, que permite al equipo madurar, crecer y

cohesionarse con el fin de mejorar desde lo técnico hasta las habilidades blandas.

Artefactos

Product Backlog

Schwaber & Sutherland (2013) afirma que el *product backlog* está compuesta por la lista de historias de usuario que se deben desarrollar en el proyecto, es decir, contiene el alcance funcional del sistema de información empresarial. Cabe resaltar que la teoría de Scrum sugiere que este documento se va afinando a medida que corren los sprint, para el caso del marco de trabajo adaptado debe quedar afinado y detallado en cada una de las etapas del ADM y sus actualizaciones se reflejarán en los artefactos de la fase de requisitos. Es importante que se identifique la prioridad y orden en el cual se va a desarrollar cada historia de usuario.

Sprint Backlog

Schwaber & Sutherland (2013) afirma que el *sprint Backlog* es una parte del *product Backlog* que se selecciona en la reunión de planeación, con el fin de ser desarrollado en el sprint que se a ejecutar. Se debe tomar las historias de usuario con mayor prioridad en el desarrollo de *software* empresarial con el fin de entregar mayor valor al usuario funcional.

6.1.6. Elicitación de requisitos

Los métodos o estrategias seleccionadas para identificar los requisitos del proyecto, deben abarcar los requisitos funcionales y los atributos de calidad, logrando así que las arquitecturas de procesos, datos, sistemas de información y tecnología cumplan con las expectativas de los interesados.

6.1.6.1. Técnicas sugeridas para la metodología

La filosofía del marco de trabajo adaptado comparte la visión de la investigación realizada por Zapata & Arango (2004), donde se centra en la alineación de los requisitos de *software* con los objetivos organizacionales y

no en problemas puntuales que busquen soluciones temporales sin analizar el panorama completo.

Teniendo en cuenta lo anterior se busca que las técnicas usadas en marco de trabajo se alejen de las técnicas tradicionales y se acerquen a los interesados, permitiendo una amplia participación del diseño e identificación de funcionalidades.

Historia de usuario

El marco de trabajo busca que los usuarios expresen sus ideas usando un lenguaje natural y plasmando sus ideas de la misma forma. Es por este motivo se usarán las historias de usuario para desglosar las necesidades que debe satisfacer el producto a desarrollar. Adicionalmente no se obliga a los involucrados a entender lenguajes de modelado que posiblemente no estén en su dominio.

Grupo ISSI (2003) define las historias de usuario como una técnica para especificar requisitos de *software* basada en tarjetas de papel que contienen las necesidades, características y requisitos funcionales y no funcionales que debe implementar el sistema de información. Izaurralde (2013) apoya esta definición y complementa que se centra en establecer conversaciones acerca de la necesidad de los usuarios, por lo que el resultado de una historia de usuario es narrado desde la perspectiva de la persona que tiene la necesidad. Beck (1999) afirma que las historias de usuario deben ser comprensibles por un programador y lo suficientemente concretas para que pueda ser implementada en pocas semanas. Yagüe & Garbajosa (2009) resalta que cada historia de usuario tiene asociado una lista de criterios de aceptación y casos de prueba para medir el cumplimiento del requisito.

Izaurralde (2013) afirma que una historia de usuario que se compone de los siguientes campos: Tipo de usuario, necesidad y justificación del desarrollo, mientras que Beck (1999) no comparte dicha afirmación ya que busca que las historias de usuario sean más completas, por lo que sugiere identificar la siguiente información: fecha, tipo de actividad, prueba funcional, consecutivo de la historia, prioridad, relación con otra historia, riesgo, estimación y notas

En el marco de trabajo adaptado se usará el siguiente estándar de historia de usuario:

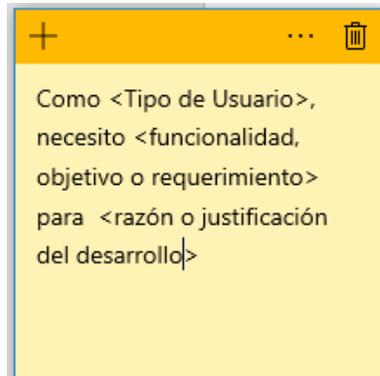


Ilustración 71. Formato de historia de usuario para el marco de trabajo adaptado

Izaurre (2013) afirma que los dueños de los productos son los encargados de garantizar que exista una lista de historias de usuario capaz de describir todo el producto a desarrollar con su respectivo orden basada en la prioridad y la generación de valor para la compañía.

Entrevistas

Teniendo en cuenta las afirmaciones que se encuentran en los siguientes párrafos el marco de trabajo adaptado busca que la comunicación verbal sea siempre una cualidad en los equipos de trabajo, ya que esta permite facilitar el conocimiento, contactar directamente a la persona con el especialista del negocio y evitar reprocesos por malos entendidos. Adicionalmente se busca que no existan jerarquías que impidan acceder al conocimiento de una persona, al contrario, los flujos de comunicación deben ser directos.

Duran & Bernardez (2002) define las entrevistas en su informe técnico como la técnica más usada en la elicitación de requisitos, siendo necesaria en cualquier momento del proceso de desarrollo de *software*. Escalona & Nora (2002) afirma que las entrevistas permiten a los analistas conocer el problema y los objetivos de la solución que se pretende implementar, adicionalmente permite que el equipo se entienda de una forma natural.

Duran & Bernardez (2002) y Pan, Zhu, & Johnson (2001) estructuran la entrevista en tres pasos: Preparación de la entrevista, realización de la entrevista y análisis de la entrevista. Estos son los mismos pasos que se van a seguir en el marco de trabajo adaptado.

Análisis de documentos

El marco de trabajo adaptado busca que los analistas revisen documentación existente de proceso de negocio sobre el cual se va desarrollar la solución empresarial. De esta forma cuando se tenga el primer contacto con los usuarios y funcionales, los términos y actividades serán familiares, adicionalmente van a facilitar la preparación de la entrevista y la redacción de preguntas puntuales sobre lo que no se entiende o se encuentra confuso. El análisis de documentos para la elicitación de requisitos es clasificado por Nuseibeh & Easterbrook (2000) como una técnica tradicional.

Design Thinking

El *Design Thinking* es una metodología que permite involucrar a los usuarios en el diseño de los productos, y mediante una serie de talleres conocer las necesidad y expectativas que debe cumplir el objeto o servicio a desarrollar. Los talleres mencionados permiten que la interacción, confianza y comunicación entre los usuarios finales y los diseñadores aumente, es por esto que *Design Thinking* hace parte de una de las técnicas de elicitación de requisitos en el marco de trabajo adaptado, ya que se pretende que los usuarios tengan una idea temprana de cómo va a quedar el sistema de información empresarial que se va a desarrollar, con el fin de evitar reprocesos mediante el descubrimiento de sus necesidades y expectativas antes de iniciar la construcción del producto, disminuyendo los riesgos en los tiempos y costos del proyecto.

Serrano & Pilar, (2014) y Gasca & Zaragoza (2014) definen *Design Thinking* como una forma para solucionar problemas reduciendo los riesgos de reproceso y las posibilidades de aumentar el éxito. Inicia enfocándose en la necesidad humana y posteriormente observa, crea prototipos y los prueba, con el fin de enlazar ideas desde las diferentes perspectivas de los involucrados.

(Ministerio de ciencia, tecnología y telecomunicaciones (Costa Rica)) generó un documento donde explica cada una de las etapas de *Design Thinking*, basados en su investigación se define cada una de ellas:

- **Empatizar.** Es el trabajo que se hace para entender y observar a los usuarios en el contexto donde se requiere el diseño. Es el esfuerzo por comprender las cosas que hacen y por qué las hacen, sus necesidades físicas y emocionales, cómo conciben el mundo y qué es significativo para los usuarios.

- **Definición.** La definición se encarga de traer claridad y enfoque del problema (entender el problema), es importante conocer bien los procesos expuestos por el usuario y su contexto. Esta etapa es sobre crear coherencia sobre las variables involucradas, procesos e información.
- **Idear.** Se inicia el proceso de diseño y generación de diversas ideas innovadoras. En este espacio se debe hacer lluvia de ideas e identificar las que más se adapten a la solución de problema. Acá se debe tener en cuenta el negocio y las capacidades tecnológicas y financieras de la empresa.
- **Prototipo.** En la etapa de prototipo se da la generación de modelos que entreguen información al equipo de diseño, estos prototipos pueden ser dibujos, artefactos, objetos, entre otros. Se debe usar material didáctico para que los diseñadores conozcan las expectativas de los usuarios. Es usado para comunicar y evitar reprocesos con costos más altos.
- **Evaluar (Testear).** En la etapa de evaluación se deben socializar los prototipos, con el fin de ser evaluados por los mismos usuarios, con el fin de afinarlos y mejorarlos. Es importante que la actitud sea de mejora continua y que no se sientan agredidos si el prototipo no es aceptado por el grupo. Lo importante es que los integrantes miren e interprete el prototipo y pueda expresar sus sentimientos sobre el mismo.

QAW

El QAW es un taller que ayuda a identificar los requisitos no funcionales que el arquitecto debe tener en cuenta para el diseño del Sistema de información empresarial, esta metodología es diseñada por el SEI (*Software Engineering Institute*). Se decidió incluir este taller en el marco de trabajo adaptado ya que todas las técnicas de elicitación de requisitos han sido enfocadas a la parte funcional y en el desarrollo de *software* es vital que los atributos de calidad sean identificados, entendidos y aprobados por el cliente, la empresa y el equipo de desarrollo de *software*.

Software Engineering Institute – SEI (2015) define el QAW o *Quality Attribute Workshops* como un método para identificar los atributos de calidad críticos o

significativos de la arquitectura como son la disponibilidad, rendimiento, seguridad, interoperabilidad, modificabilidad, entre otros. Adicionalmente resalta que los desafíos más grandes que enfrenta el taller es determinar qué atributos de calidad son importantes antes de construir el sistema, conocer las necesidades y expectativas de las partes interesadas de forma organizada y efectiva y saber cómo mejorar y aumentar la comunicación entre las partes interesadas.

En el desarrollo del QAW se espera que el grupo sea multifuncional, es decir, contar con personas del negocio (Funcionales y directivos), personas de TI del cliente y arquitectos o persona de TI que representen al ente que va a desarrollar el sistema de información.

Software Engineering Institute – SEI (2015) divide el QAW en los siguientes pasos:

- **Presentación e introducción del QAW:** Este paso se refiere a la socialización del taller, la explicación de cada paso, el objetivo de su ejecución y cómo cada uno de los integrantes participa de él.
- **Presentación del negocio:** Un participante del negocio socializa cuál es el objetivo del proceso afectado con el proyecto al interior de la compañía.
- **Presentación del plan arquitectónico:** Un interesado técnico expone la arquitectura a alto nivel que se ha levantado basado en documentos previamente estudiados de la situación tecnológica de la compañía donde se va a desarrollar el proyecto.
- **Identificación de conductores arquitectónicos:** Los conductores arquitectónicos están compuestos por requisitos funcionales de alto nivel, principalmente los arquitectónicamente significativos, restricciones y requisitos no funcionales de alto nivel. En este paso el grupo llega a un consenso para determinar cuáles son claves para el sistema.
- **Identificación de escenarios de calidad:** Los interesados crean escenarios del mundo real, respecto al proceso que será afectado con el sistema de información a desarrollar, con el fin de garantizar que la arquitectura y el diseño del *software* cumpla con estos. Los escenarios están compuestos por un estímulo, una condición ambiental y una respuesta.

- Consolidación de escenarios: Se agrupan los escenarios de calidad que se encuentren relacionados o repetidos.
- Priorización de escenarios: Los interesados priorizan los escenarios de calidad a través de un proceso de votación, teniendo en cuenta las necesidades del sistema de información empresarial que se va a desarrollar.
- Refinamiento de los escenarios de calidad: los escenarios con mayor prioridad son tomados para completarlos en caso de que falte información o aclararlos.

6.2. Objetivo 2 – Marco de trabajo adaptado

En el marco de trabajo adaptado se va a explicar el objetivo de cada fase, las actividades o talleres que se deben realizar y los artefactos que se deben de generar, cabe resaltar que algunos de los artefactos son el producto de la realización de actividades o talleres ejecutados con los clientes. Adicionalmente, todas las actividades se centran en el desarrollo de aplicaciones empresariales.

La figura muestra que el ADM es secuencial, sin embargo, se aclara que es posible moverse entre diferentes fases con el fin de afinar y mejorar el diseño de las arquitecturas.

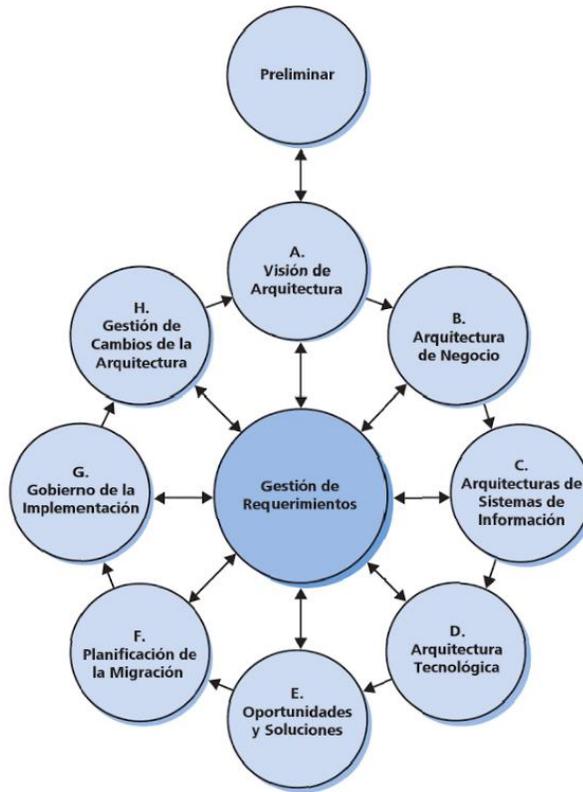


Ilustración 72. Ciclo del método de desarrollo de arquitectura ⁷⁰

Nota aclaratoria: Las plantillas para diagramar la arquitectura fueron construidas usando la aplicación de *Enterprise Architect*, sin embargo, la implementación de la metodología se puede realizar sobre cualquier herramienta que permite la diagramación y trazabilidad de los componentes. La aplicación cuenta con un trial por 30 días que puede ser descargado de la página del fabricante⁷¹ o en caso de no querer descargar la aplicación se tienen versiones en HTML, las cuales pueden ser visualizadas usando un explorador.

Al final de cada fase se adicionan las evidencias de la prueba piloto que se realizó para aplicar el marco de trabajo adaptado. Para mayor información de la prueba piloto ver la sección de [Objetivo 3- Implementación del marco de trabajo adaptado](#)

⁷⁰ Tomado de (Josey, 2013)

⁷¹ <http://www.sparxsystems.com/products/ea/trial/request.html>

6.2.1. Fase preliminar

6.2.1.1. Generalidades de la fase

The Open Group (2011) afirma que el objetivo de la fase es preparar a la organización para iniciar proyectos de arquitectura empresarial, adicionalmente se trata de definir, dónde, qué, porqué, quién y cómo se va a ejecutar el proyecto.

En el marco de trabajo adaptado para el desarrollo de aplicaciones empresariales se busca identificar cuál es el objetivo de la aplicación, qué procesos se verán afectados con el nuevo desarrollo, identificación de los participantes y comunicación oficial del inicio del proyecto.

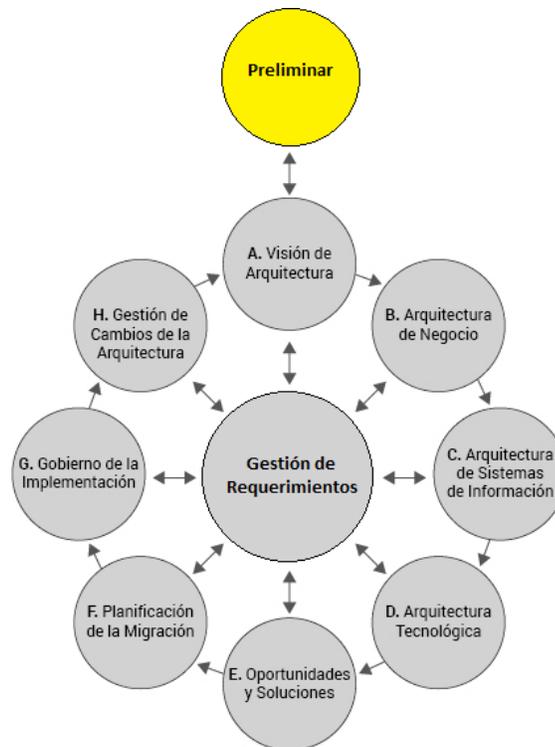


Ilustración 73. Fase preliminar⁷²

⁷² Tomado de (The Open Group, 2011)

6.2.1.2. Actividades para ejecutar

Reunión con los directivos para conocer sus motivaciones

La reunión con los directos o las personas que cuentan con poder para tomar las decisiones sirve para conocer las expectativas y hacia dónde quieren enfocar los productos o servicios que se van a desarrollar mediante el desarrollo de la arquitectura. Es muy importante que en esta reunión se identifiquen los objetivos estratégicos que se están impactando con el desarrollo del sistema de información empresarial, ya que la base del marco de trabajo adaptado es la alineación de la estrategia con los desarrollos tecnológicos. Adicionalmente esta relación justifica el desarrollo de la arquitectura y garantiza que la inversión está directamente relacionada con la estrategia.

Adicionalmente se debe identificar cuáles son los procesos impactados por la ejecución del proyecto, con el fin de identificar todos los interesados en el desarrollo del sistema de información empresarial, esto permite la clasificación de los mismos e identificar a cuáles talleres o reuniones a deben ser convocados. Cabe resaltar que esta clasificación e identificación es muy temprana y se deben hacer trabajos adicionales para la gestión de interesados, sin embargo, permite al arquitecto dimensionar las personas que estarán involucradas en el desarrollo de la arquitectura.

Se recomienda preparar una entrevista para identificar la siguiente información y diligenciar el formato de declaración del trabajo de arquitectura (Artefacto generado de esta reunión).

- Organigrama de la compañía con fin de identificar las áreas y procesos afectados
- Objetivos estratégicos para mapearlos con las necesidades a alto nivel del sistema de información empresarial
- Lista de interesados y roles al interior de la compañía
- Motivaciones para desarrollar el sistema de información empresarial
- Objetivo del desarrollo del sistema de información empresarial
- Principios de negocio
- Principios de arquitectura

Reunión de Socialización del marco de trabajo adaptado

Esta reunión se realizará con todos los interesados que fueron identificados en la *reunión con los directivos para conocer sus motivaciones*. El objetivo principal es que todos los integrantes conozcan cómo se va a trabajar y qué papel juegan al interior del proyecto. Se considera de alta importancia esta reunión ya que se tendrá una visión y un panorama general del marco de trabajo adaptado.

Para apoyar esta reunión se debe usar la presentación “*Presentación marco de trabajo.pptx*” ubicado en el repositorio de arquitectura en la carpeta presentaciones.

Solicitar y revisar de documentos de arquitectura existentes

Esta actividad servirá para identificar si la compañía ya tiene documentos existentes que aporten al desarrollo de la arquitectura como son: principios de arquitectura, lineamientos, estándares, entre otros. Con el fin de conocer un poco más sobre el gobierno y aspectos para tener en cuenta en el desarrollo de la arquitectura empresarial. Adicionalmente el marco de trabajo propone unos principios de arquitectura los cuales pueden ser completados o reemplazados con la información entregada por la compañía. El documento de *Principios.docx* se encuentra en el repositorio de arquitectura en la carpeta documentación de apoyo.

Solicitar y revisar documentación de los procesos afectados

Conocer los procesos es vital en el desarrollo de arquitectura empresarial, ya que estos están diseñados para cumplir el plan estratégico de la compañía. La documentación de procesos debe ser solicitada y estudiada con el fin de conocer como tienen diseñados los procesos y la interacción que tiene entre ellos. Cabe resaltar que esta actividad se debe hacer antes de tener algún contacto con los clientes, con el fin de entender el proceso, conocer los detalles, lenguaje y llevar preguntas puntuales para mejorar su entendimiento. Adicionalmente se resalta que en los clientes causará una buena impresión que se interesen por su proceso.

En caso de que la compañía no cuente con el diseño de procesos es muy importante hacer un alto en el desarrollo de la arquitectura empresarial e

informarle a los directivos y patrocinador que es necesario tener el proceso definido antes de sistematizarlo, ya que si se continúa, el riesgo de que los requisitos no estén basados en la necesidad de la compañía sino en los caprichos del usuario aumenta, lo que puede generar cambios según el punto de vista del usuario que lo revise y estos cambios se traducen en dinero y reprocesos. Si los procesos no están definidos se debe programar el tiempo para desarrollar dicha actividad.

6.2.1.3. Artefactos generados

Declaración del trabajo de arquitectura

La declaración del trabajo de arquitectura es el artefacto generado en la reunión de los directivos para conocer sus motivaciones. Esta es la cédula del proyecto para el desarrollo de la arquitectura que tendrá como resultado una aplicación empresarial.

Cabe resaltar que la información puede ser diligenciada completamente si se cumple con un primer objetivo del marco de trabajo adaptado y es que el desarrollo de la arquitectura empresarial ayude a cumplir uno o varios objetivos estratégicos, de lo contrario no se podría garantizar que la inversión en el proyecto satisfaga el cumplimiento de la estrategia.

El formato de la declaración del trabajo de arquitectura se encuentra en el repositorio de arquitectura en la carpeta plantillas, bajo el nombre de "Declaración del trabajo de arquitectura.doc". Cuando este formato haya sido diligenciado se debe almacenar en el repositorio de arquitectura en la siguiente ruta: Arquitecturas → Arquitectura a trabajar → Documentos de gestión

Principios de arquitectura

The Open Group (2011) define un principio como una regla o pauta destinada a ser duradera y pocas veces modificadas, su principal objetivo es apoyar y servir de guía para que una organización cumpla con su misión. Los principios de la arquitectura deben ser claramente trazables y articulados para guiar la toma de decisiones. Son elegidos para asegurar la alineación de la arquitectura y la implementación de la arquitectura de destino con estrategias de negocios.

“*Principios de arquitectura.docx*” es un documento entregado en el marco de trabajo como referencia para para el desarrollo de proyectos, cabe resaltar que la elaboración de este documento se basó en los principios sugeridos por The Open Group (2011). Sin embargo, se debe socializar para garantizar que la compañía esté de acuerdo o complementarse con la información solicitada en la actividad de “*Solicitar y revisar de documentos de arquitectura existentes*”. El documento “*Principios de arquitectura.docx*” se encuentra en el repositorio de arquitectura en la carpeta de documentos de apoyo.

6.2.1.4. Documentos generados en el piloto

Declaración de trabajo de arquitectura

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
Declaracion del trabajo de arquitectura.docx	yo	19 feb. 2018 yo	124 KB

Ilustración 74. Evidencia de la declaración del trabajo de arquitectura

6.2.2. Fase A - Visión de la arquitectura

6.2.2.1. Generalidades de la fase

The Open Group (2011) afirma que la fase A “*Visión de la arquitectura*”, define una primera definición del alcance del proyecto, es decir, lo que está incluido y lo que no está incluido, adicionalmente se define las limitaciones y las expectativas de los interesados. Cabe resaltar que la definición del alcance debe ser realista y basarse en las capacidades, recursos y competencias que tiene la empresa, por ejemplo, no se puede pretender montar un bus de servicios IBM si todas las plataformas de la empresa son Microsoft y no se tiene presupuesto para adquisiciones de infraestructura. Cabe resaltar que en las definiciones se debe tener en cuenta los siguientes aspectos:

- El alcance que se va a definir debe estar relacionado directamente con los objetivos específicos del proyecto.
- Las restricciones se definirán basados en los principios de arquitectura desarrollados en la Fase Preliminar.

The Open Group (2011) resalta la importancia de la visión de la arquitectura, ya que proporciona al arquitecto una herramienta para presentar los beneficios que generará el desarrollo de la arquitectura, y de esta forma dar a conocer a los interesados y patrocinadores la importancia de su desarrollo. La visión de la arquitectura describe cómo la nueva capacidad alcanzará los objetivos empresariales y los objetivos estratégicos.

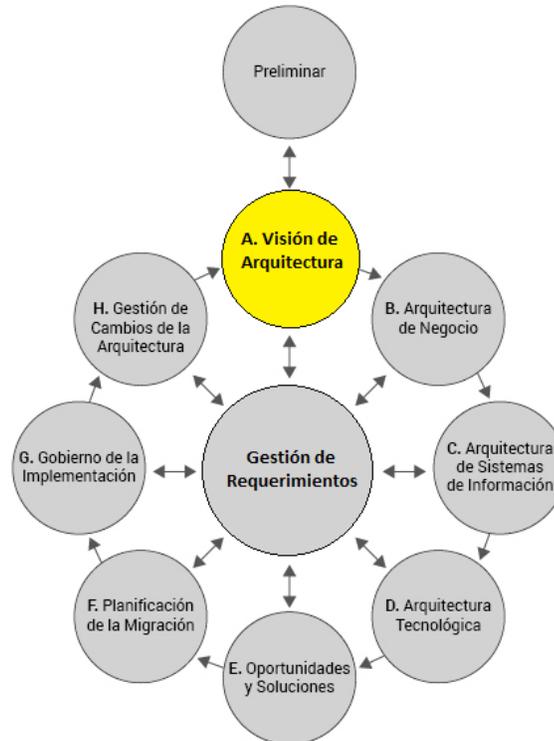


Ilustración 75. Fase A - Visión de la solución⁷³

6.2.2.2. Actividades para ejecutar

Buscar la aprobación de la declaración del trabajo de arquitectura

Como se mencionó en la fase preliminar, la declaración de la arquitectura busca que el proyecto de arquitectura sea oficial al interior de la compañía, sin embargo, es importante que este documento esté firmado por el patrocinador, con el fin de garantizar que el proyecto está aprobado y tiene presupuesto para

⁷³ Tomado de (The Open Group, 2011)

la ejecución. *(Ver detalles del documento “Declaración del trabajo de arquitectura aprobado” en la sección de artefactos entregados)*

Entrevistar involucrados para identificar la visión del a solución

En la visión de la solución se busca dejar claros puntos específicos, con el fin de que todos conozcan el alcance, restricciones y expectativas de cada uno de los involucrados. Para esto es importante hacer una serie de sesiones con diferente grupo de persona e identificar la siguiente información:

- Objetivos específicos
- Alcance de la solución
- Preocupaciones y requerimientos de los interesados identificados en la fase preliminar
- Limitaciones o restricciones
- Riesgos
- Capacidades
- Plan de comunicación

(Ver detalles del documento “Visión de la arquitectura” en la sección de artefactos entregados)

Reunión para socializar la declaración de trabajo de arquitectura y visión

Se debe tener una reunión para socializar el resultado de la fase A: Visión de arquitectura, con el fin de que todos los involucrados conozcan el alcance y los aspectos que se van a mejorar con el desarrollo de la arquitectura. Es importante que esta reunión sea clara y concisa. Adicionalmente se debe validar que el mensaje se haya entendido por todos los participantes de la reunión.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es

posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. (*Ver detalles del “Acta” en la sección de artefactos entregados*)

6.2.2.3. Artefactos generados

Declaración del trabajo de arquitectura aprobado

La definición del entregable se encuentra detallada en una fase anterior, para consultarla hacer click en el siguiente enlace: [Declaración del trabajo de arquitectura](#)

Cabe resaltar que al finalizar la fase A “*visión de la arquitectura*” este documento debe estar completamente diligenciado y terminado, listo para la firma del patrocinador.

Visión de la arquitectura

El documento de visión de la arquitectura expone el alcance, procesos, sistemas de información e interesados impactados con el sistema de información empresarial. Adicionalmente se evidencian los beneficios del desarrollo del sistema de información empresarial y teniendo la declaración del trabajo de arquitectura como complemento, se puede visualizar la trazabilidad con a la estrategia.

El formato de la visión de la arquitectura se encuentra en el repositorio de arquitectura en la carpeta plantillas → Visión de arquitectura.docx

El formato de la visión de arquitectura se encuentra en el repositorio de arquitectura en la carpeta de plantillas, bajo el nombre de “Vision de arquitectura.docx”. Cuando este formato haya sido diligenciado se debe almacenar en repositorio de arquitectura, en la carpeta Arquitectura → Arquitectura a trabajar → Documentos de gestión → Vision de arquitectura.docx.

Principios de arquitectura

En la fase A “*visión de la arquitectura*” se busca que este documento se valide para identificar si está actualizado o hacer algunas modificaciones.

El documento de declaración de trabajo de arquitectura diligenciado se encuentra en el repositorio de arquitectura, en la carpeta documentos de apoyo → Principios de arquitectura.docx. Este debe ser leído y adaptado a la necesidades, políticas y restricciones de la compañía

Definición de arquitectura

Los componentes de la motivación serán desarrollados sobre el archivo “*Definición de arquitectura.EAP*” en la sección de arquitectura base y arquitectura destino. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP. Este diagrama se modelará en la sección motivación del archivo.

El diagrama de motivación se puede consultar la sección de [Motivación](#) En esta sección se explica cada uno de los componentes requeridos para diagramar y su relación con otros componentes y capas.

Acta

Documento donde se consignan los temas tratados en una reunión del equipo de trabajo, con el fin de dejar evidencia de los temas tratados, compromisos y aprobaciones realizadas. El formato del acta se encuentra en el repositorio de arquitectura en la carpeta de plantillas, bajo el nombre de “Acta de Reunión.docx”. Cuando este formato haya sido diligenciado y firmado se debe almacenar en repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Actas.

6.2.2.4. Documentos generados en el piloto

Visión de la arquitectura



Buscar en Drive

Mi u... > Trabaj... > Entr... > Repositorio de ... > Arquit... > Arquitectura ... > Documentos de Gesti...

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
W Declaracion del trabajo de arquitectura.docx	yo	19 feb. 2018 yo	124 KB
W Plan Implementacion y Migracion.docx	yo	3 mar. 2018 yo	88 KB
W Visión de arquitectura.docx	yo	19 feb. 2018 yo	79 KB

Ilustración 76. Evidencia Visión de la arquitectura

Principios de arquitectura



Mi unidad > TrabajoGrado > Entregable > Repositorio de arquitectura > Principios

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
W Principios de arquitectura.docx	yo	19 feb. 2018 yo	77 KB

Ilustración 77. Evidencia de principios

Actas



Mi un... > Trabajo... > Entreg... > Repositorio de arqu... > Arquitect... > Arquitectura Ca... > Act...

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
W Acta de Reunion_ArquitecturaAplicaciones.docx	yo	16:57 yo	47 KB
W Acta de Reunion_ArquitecturaNegocio.docx	yo	16:57 yo	47 KB
W Acta de Reunion_ArquitecturaTecnologica.docx	yo	16:57 yo	47 KB
W Acta de Reunion_Control de cambio.docx	yo	16:57 yo	47 KB
W Acta de Reunion_Oportunidades_Soluciones.docx	yo	16:57 yo	47 KB
W Acta de Reunion_Planificar_Migracion.docx	yo	16:57 yo	47 KB
W Acta de Reunion_Requisitos.docx	yo	16:57 yo	47 KB
W Acta de Reunion_Visión.docx	yo	16:57 yo	47 KB

Ilustración 78. Evidencia de las actas de aprobación

Diagrama de motivación

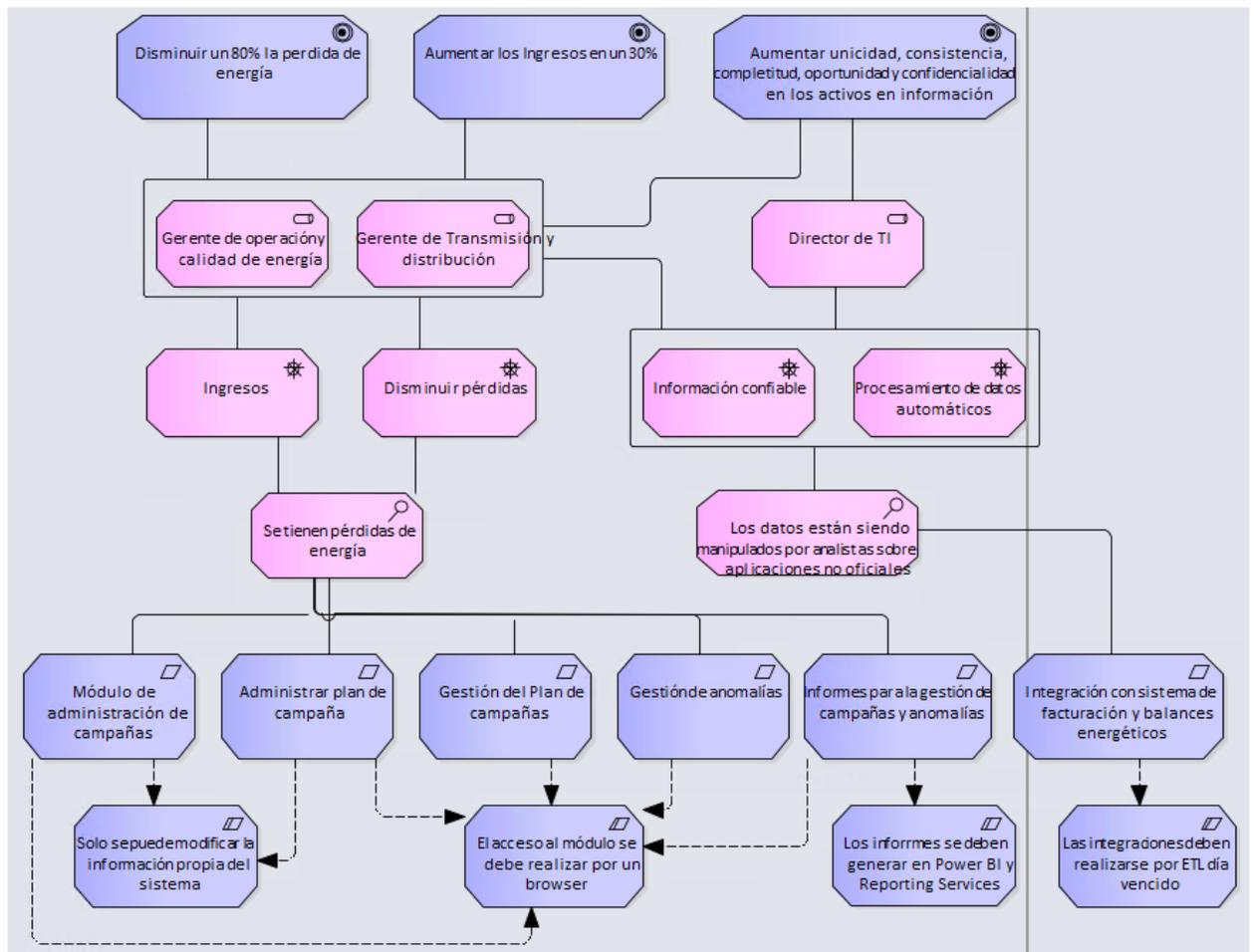


Ilustración 79. Evidencia del diagrama de motivación

6.2.3. Fase B - Arquitectura del negocio

6.2.3.1. Generalidades de la fase

The Open Group (2011) define la arquitectura de negocio como el diseño de la estrategia de producto o servicio, adicionalmente también describe procesos organizacionales que detallan cómo se alcanzarán los objetivos estratégicos definidos por la empresa.

En esta fase es muy importante estudiar detalladamente la documentación de proceso que fueron solicitados en la fase preliminar, con el fin de entenderlos e identificar en qué formar serán afectados con el desarrollo del sistema de información empresarial. Adicionalmente en las entrevistas con los interesados

se debe identificar las aplicaciones que actualmente son usadas para soportar el proceso.

Cabe resaltar, que si los procesos de negocio no están definidos se debe hacer un alto en el camino para informar al patrocinados que se requieren realizar actividades del levantamiento de procesos, las cuales deben ser validadas frente a la estrategia. Este insumo es vital para continuar con el análisis de las otras arquitecturas. Sin este insumo se pierde el foco del marco de trabajo adaptado, ya que este busca la alineación de los sistemas de información empresarial con la estrategia de la compañía y se desarrollarían funcionalidades que solo están avaladas por usuario finales y aún no han sido socializadas, ni aprobadas.

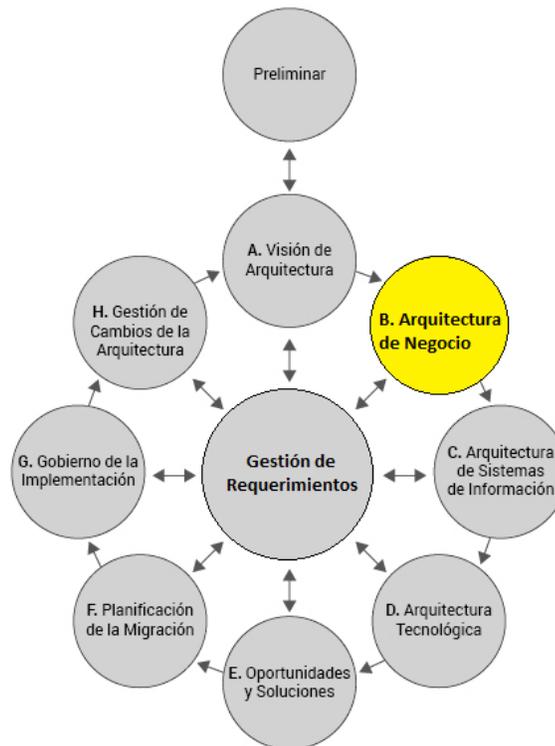


Ilustración 80. Fase B - Arquitectura del Negocio⁷⁴

⁷⁴ Tomado de (The Open Group, 2011)

6.2.3.2. Actividades para ejecutar

Estudiar y entender los procesos implicados en el proyecto

Esta actividad busca entender detalladamente los procesos implicados en el desarrollo del sistema de información empresarial y se busca identificar de cada proceso los siguientes componentes:

- Actores
- Roles
- Colaboraciones
- Disparadores
- Servicios
- Interfaces
- Productos
- Relación con otros procesos
- Información o artefactos de entrada
- Información o artefactos de salida

(Ver detalles de la “Arquitectura de negocio” en la sección de artefactos entregados)

Levantar requisitos del sistema de información empresarial

El levantamiento de requisitos es una actividad transversal, ya que en cada una de las fases se identifican y afinan requerimientos del sistema de información empresarial. Estos requisitos son clasificados en: requisitos funcionales, requisitos de información, reglas de negocio y atributos de calidad. Los requerimientos detallan las historias de usuario que los clientes identifican en los talleres. Por lo tanto, es muy importante identificar las historias de usuario y por cada una de ellas las reglas de negocio y requisitos de información que se deben de tener en cuenta.

Para realizar la toma de requisitos es importante reunirse con los funcionales o las personas que operan día a día el proceso. Cabe resaltar que todos los requisitos deben mapearse con las actividades definidas en la documentación de procesos.

Cabe resaltar que todos los requisitos deben tener asociado una historia de usuario, con el fin de tener trazabilidad en las funcionalidades que se van a

desarrollar. (Ver detalles de la “Arquitectura de negocio e historias de usuario” en la sección de artefactos entregados)

Diagrama de arquitectura de negocio

Los componentes identificados en la actividad “Estudiar y entender los procesos implicados en el proyecto” deben ser diagramados con Archimate 3.0, en la sección de arquitectura base y arquitectura destino. Este diagrama tendrá una única versión que representará la arquitectura de negocio actual y la arquitectura de negocio destino, aunque es un insumo primordial para el desarrollo del sistema de información empresarial, solo se diagramará lo que se está ejecutando, es decir, la arquitectura actual.

Si se requiere detallar los procesos de negocio se deben modelar usando el lenguaje BPMN 2.0, con este diagrama se puede visualizar los flujos, decisiones, entradas, salidas, entre otros. (Ver detalles de la “Arquitectura de negocio” en la sección de artefactos entregados)

Análisis de información

A medida que se toman requisitos y se tiene contacto con los involucrados se identifican nuevos documentos o formatos de apoyo que son usados en los procesos y procedimientos del cliente, dichos documentos deben ser analizados y estudiados por los arquitectos con el fin de identificar información que el usuario no haya expresado en las sesiones de requisitos. Esta es una actividad que se realiza continuamente, con el fin de identificar detalles que los suelen omitir, ya que lo consideran obvio o asumen que es del dominio del arquitecto.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta

firmada por el responsable de la aprobación. (*Ver detalles del “Acta” en la sección de artefactos entregados*)

6.2.3.3. Artefactos generados

Arquitectura del negocio

El diagrama de arquitectura de negocio será desarrollado sobre el archivo “*Definición de arquitectura.EAP*” en la sección de arquitectura base y arquitectura destino. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectuta.EAP.

Este diagrama tendrá una única versión que representará la arquitectura de negocio actual y la arquitectura de negocio destino, aunque es un insumo primordial para el desarrollo del sistema de información empresarial, solo se diagramará lo que se está ejecutando.

La diagramación en Archimate 3.0 o el detalle de los procesos en BPMN 2.0 se encuentran en la sección [Archimate](#) y [BPMN](#) respectivamente.

Historias de usuario

Las historias de los usuarios serán desarrolladas sobre el archivo “*Definición de arquitectuta.EAP*” en la sección de historia de usuario. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitecturas → Arquitectura a trabajar → Definición de arquitectuta.EAP.

En el marco de trabajo adaptado se usará el siguiente estándar de historia de usuario:

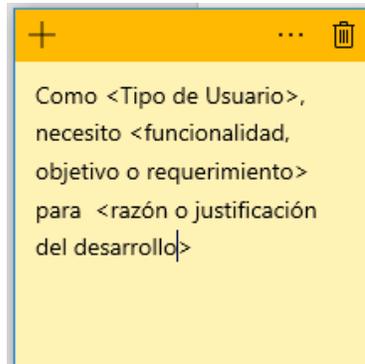


Ilustración 81. Formato de las historias de usuario para el marco de trabajo adaptado

Producto backlog

El *product Backlog* es la lista de historias de usuarios priorizada. Se almacenará en el archivo “*Definición de arquietctura.EAP*” en la sección de historia de usuario, sin embargo, se usarán agrupadores para indicar a qué modulo del sistema de información empresarial pertenece y permita identificar su prioridad.

Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitecturas → Arquitectura a trabajar → Definición de arquietctura.EAP.

Requerimientos

En este punto del desarrollo de la arquitectura los requisitos serán tomados usando la técnica de entrevistas, las preguntas nacen a partir del estudio y conocimiento que se tiene del proceso y los documentos estudiados previamente antes de reunirse con los usuarios funcionales.

Los requisitos se deben consignar en el documento de Definición de arquitectura, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitecturas → Arquitectura a trabajar → Definición de arquitectura.EAP.

Catálogo de servicios

Aranda *software* (2010) define un catálogo de servicios como una provisión formal de servicios entre TI y los clientes. En el marco de trabajo adaptado el catálogo de servicios es un documento estándar y único para toda la empresa, debe ser alimentado en cada uno de los proyectos donde se pueda evidenciar cuáles son los servicios que presta la compañía a nivel de negocio, aplicaciones e infraestructura y que sistemas de información o actores los consumen.

Con lo anterior se puede tener una trazabilidad de cada uno de los servicios e identificar cómo afecta la modificación de alguno de ellos al interior de la compañía.

El catálogo de servicios se debe consignar en el documento de Catalogo de Arquitectura, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Catálogos → Catálogo de arquitectura.EAP.

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.3.4. Documentos generados en el piloto

Historias de usuario

HU-001 Administración grupo circuito	HU-015 Carga de planes circuito por semana
HU-002 Administración de tipos de observación	HU-016 Administración del plan de campañas
HU-003 Administración de tipos de anomalías	HU-017 Observaciones del plan de campaña
HU-004 Administración de tipo de reporte	HU-018 Avance automático al plan de campañas
HU-005 Administración de clase reporte	HU-019 Avances manuales al plan de campañas
HU-006 Administración de tipo de ordenes de servicio	HU-020 Gestión del plan de campañas
HU-007 Administración de acciones	HU-021 Informe de evolución plan de campaña
HU-008 Administración de justificaciones de cierre forzado	
HU-009 Administración de tipo de acciones	HU-022 Creación de anomalías de forma automática
HU-010 Administración de pesos por tarifa	HU-023 Registro de anomalías manualmente
HU-011 Administración de pesos por circuito	HU-024 Cierre de anomalías manualmente
HU-012 Administración de tareas del plan de campaña	HU-025 Interpretación de observaciones de las OS
HU-013 Administración de metas por tipo de macromedidores	HU-026 Validación de anomalías repetidas
HU-014 Administración de anomalías lector open	HU-027 Carga masiva de anomalías
	HU-028 Modificación masiva de anomalías

Ilustración 82. Evidencia de las historias de usuario

6.2.4. Fase C - Arquitectura de sistemas de información

6.2.4.1. Generalidades de la fase

The Open Group (2011) en esta fase se trabaja una combinación de datos y arquitectura de aplicaciones, donde se identifica cuáles son las aplicaciones que soportan la sistematización de los procesos y se identifican las entidades del diagrama de dominio. Con la información anterior se puede identificar que aplicaciones continúan y cuáles serán afectada o eliminadas de la compañía

a raíz del nuevo sistema de información empresarial que se va a implementar. Adicionalmente se conocen cuáles son las entidades de negocio que se van a incluir en el desarrollo e identificar si son propias del sistema o serán proporcionados por otros sistemas de información, es decir, si identifica el gobierno de los mismo.

El resultado principal de esta fase es diagramar la arquitectura de sistemas de información y datos actual, la arquitectura destino y las brechas identificadas para llegar a la arquitectura objetivo.

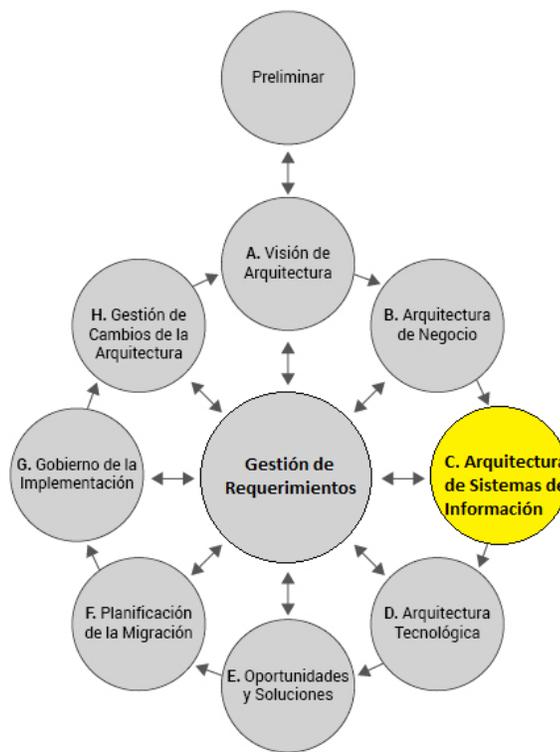


Ilustración 83. Fase C - Arquitectura de sistemas de Información⁷⁵

6.2.4.2. Actividades para ejecutar

Entrevistas a los usuarios funcionales

La entrevista con los usuarios funcionales permite la identificación de los sistemas de información o herramientas que usan para el procesamiento de datos en las áreas que hacen parte del alcance del proyecto, es decir, puede

⁷⁵ Tomado de (The Open Group, 2011)

ser desde un ERP hasta un archivo en Excel. Cabe resaltar que cuando una aplicación es muy grande es importante identificar el módulo en particular que se está usando.

El arquitecto debe ser capaz de identificar cuál de estas aplicaciones será eliminada o intervenida para llegar al a arquitectura destino. Adicionalmente con esta información también se debe identificar la arquitectura base de aplicaciones y datos. En estas entrevistas es muy importante identificar las entidades de negocio con el fin de saber si son manejadas directamente desde el sistema de información o serán tomadas de otras fuentes de datos.

Para capturar la data de estas entrevistas se puede usar el formato “Aplicaciones y Datos.xls” ubicado en el repositorio de arquitectura en la carpeta de plantillas.

Taller de *Desing Thinking*

El taller de *Design Thinking* permite tener un acercamiento a los usuarios funcionales para conocer sus expectativas y lo que esperan del producto. El resultado de este taller son los prototipos de la aplicación diseñados por los mismos usuarios. Antes de iniciar el taller se debe presentar los pasos de los cuales está compuesto y el rol que cada uno de los integrantes cumple. Se resalta que la persona que lleva el taller debe empatizar con los integrantes, con el fin de generarles confianza y les permita su libre desarrollo y confianza para expresar sus ideas.

La presentación que se tendrá de apoyo para este taller se encuentra en el repositorio de arquitectura en la capeta presentaciones bajo el nombre de “*Design Thinking.ppt*”

Cabe resaltar que si en estos talleres si identifican requisitos funcionales o no funcionales se deben consignar en el archivo “Definición de arquitectura.EAP” en la sección de requisitos funcionales.

La documentación para desarrollar los talleres de *Design Thinkin* se puede consultar en [Design Thinking](#) En esta sección se explica cada uno de los pasos que se deben desarrollar en la ejecución del taller.

Taller QAW

El QAW o taller para identificar atributos de calidad permite a los arquitectos identificar requerimientos no funcionales expresados por los mismos usuarios. De esta forma se tiene las bases necesarias para tomar decisiones de arquitectura y realizar el diseño del sistema de información empresarial.

Antes de iniciar el taller se debe hacer una presentación de la metodología y explicar cada uno de los pasos, esta presentación explica cómo se debe trabajar, qué pasos seguir y qué formatos llenar. El archivo que se tendrá de apoyo para este taller se encuentra en el repositorio de arquitectura en la carpeta presentaciones bajo el nombre de “*Taller QAW.ppt*”

En la ejecución del taller se deben llenar unos formatos para conocer los escenarios de calidad que plantean los participantes, este formato se encuentra en el repositorio de arquitectura en la carpeta plantillas bajo el nombre de “*Escenario de Calidad.docx*”

Por último, esta información debe ser consolidada, clasificada en atributos de calidad y consignada en el archivo “*Definición de arquitectura.EAP*” en la sección de requisitos no funcionales.

En caso de tener dudas sobre la metodología QAW revisar la documentación de apoyo en el capítulo [QAW](#). En esta sección se explica cada uno de los pasos que se deben desarrollar en la ejecución del taller y las personas que deben asistir con el fin de garantizar su correcto desarrollo.

Análisis de información

A medida que se toman requisitos y se tiene contacto con los involucrados se identifican nuevos documentos o formatos de apoyo, los cuales deben ser analizados y estudiados por los arquitectos con el fin de identificar información que el usuario no haya expresado en las sesiones de requisitos. Esta es una actividad que se realiza continuamente, con el fin de identificar detalles que los suelen omitir, ya que lo consideran obvio o asumen que es del dominio del arquitecto.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. (*Ver detalles del “Acta” en la sección de artefactos entregados*)

6.2.4.3. Artefactos generados

Arquitectura de línea base de sistemas de información y datos

La arquitectura base permite la visualización del estado actual de los sistemas de información y datos de la compañía. Cabe resaltar que solo se diagramarán los componentes de aplicaciones y datos que intervienen en los procesos afectados por el desarrollo del proyecto. Este diagrama contiene los sistemas de información, documentos, plantillas, entre otros; que actualmente usa el proceso para sistematizar los datos.

La arquitectura base de las aplicaciones y datos será desarrollado sobre el archivo “*Definición de arquitectura.EAP*” en la sección de arquitectura base. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP.

La diagramación de la arquitectura de Aplicaciones y datos en Archimate 3.0 se encuentra en la sección [Capa de Aplicaciones](#). Allí se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Arquitectura destino de sistemas de información y datos

La arquitectura de destino permite a los arquitectos diagramar el punto donde se quiere llegar con el desarrollo de la misma. En el repositorio de arquitectura,

en la carpeta Arquitectura de Referencia se encuentran algunos modelos de referencia que ayudan y facilitan el diseño del sistema de información empresarial. Este diagrama contiene los sistemas de información y relación entre ellos que serán usados por el proceso para alcanzar los objetivos corporativos.

La arquitectura de destino de las aplicaciones y datos será desarrollada sobre el archivo "*Definición de arquitectura.EAP*" en la sección de arquitectura destino. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP.

La diagramación de la arquitectura de Aplicaciones y datos en Archimate 3.0 revisar se encuentra en la sección [Capa de Aplicaciones](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Brechas

Las brechas son los paquetes de trabajo o desarrollos que se deben ejecutar para alcanzar la arquitectura destino desde la arquitectura base. Es decir, especifica qué funcionalidades, módulos, servicios, entre otros se deben desarrollar para llegar al punto deseado por el cliente para lograr los objetivos corporativos.

Las brechas serán modeladas sobre el archivo "*Definición de arquitectura.EAP*" en la sección de Análisis de brechas. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP.

La diagramación de las las brechas en Archimate 3.0 se encuentra en la sección [Implementación y migración](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Historias de usuario

La definición de las historias de usuario y espacio de almacenamiento se puede consultar en la sección de [Historia de Usuario](#)

Prototipos

Los prototipos son el resultado del taller de *Design Thinking*, el objetivo de los prototipos es que los usuarios que van a operar el sistema de información empresarial diseñen cómo serán las pantallas y la navegación del mismo. De esta forma se garantiza que el *software* sea aceptado y mejore la experiencia de usuario.

Como los prototipos fueron desarrollados en el taller de *Design Thinking* se puede tener evidencia de las decisiones de los usuarios usando fotografías, sin embargo, es importante pasarlos a limpio para ser entregadas posteriormente al equipo de desarrollo del producto. Existen varias herramientas para generar los prototipos, sin embargo, el marco de trabajo adapta no generará sesgos, por lo que la elección de la herramienta será de libre elección.

Los prototipos deben almacenarse en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Prototipos

Requisitos funcionales

En este punto del desarrollo del proyecto los requisitos serán tomados usando la técnica de entrevistas, las preguntas nacen a partir del estudio y conocimiento que se tiene del proceso y los documentos estudiados previamente antes de reunirse con los usuarios funcionales. Para realizar la toma de requisitos es importante reunirse con los funcionales o las personas que operan día a día el proceso. Cabe resaltar que todos los requisitos deben mapearse con las actividades definidas en la documentación de los procesos.

Cabe resaltar que todos los requisitos deben tener asociado una historia de usuario, con el fin de tener trazabilidad en las funcionalidades que se van a desarrollar.

Los requisitos funcionales se deben consignar en el documento de Definición de arquitectura, en la sección de requisitos funcionales, esta plantilla se

encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Requisitos → Requisitos Funcionales

Requisitos no funcionales

Los requisitos no funcionales o atributo de calidad son el resultado del desarrollo del taller QAW. Este taller se basa en escenarios de calidad identificado por los usuarios y posteriormente convertidos en requisitos no funcionales medibles. Con la identificación de los requisitos de calidad los arquitectos tienen las bases necesarias para realizar el diseño del sistema de información y tomar las decisiones arquitectónicas con bases sólidas.

Los requisitos no funcionales se deben consignar en el documento de Definición de arquitectura, en la sección de requisitos no funcionales, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Requisitos → Requisitos No Funcionales

Modelo de dominio

El modelo de dominio se debe consignar en el documento de Definición de arquitectura, en la sección diagrama de dominio, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Sección de diseño → Modelo de dominio

Modelo entidad relación

El modelo entidad relación nace de las entidades del modelo de dominio, sin embargo, en esta sección se debe tener en cuenta que este modelo es físico por lo que se debe tener claro el nombre de las tablas, atributos, tipos de datos, entre otros. Adicionalmente se debe identificar qué tablas contienen información transaccional o datos maestros y saber si estos datos son

administrados desde el sistema de información empresarial a desarrollar o es cargado directamente de otras fuentes de datos.

Para consultar el modelo entidad relación se debe hacer desde el motor de base de datos que se haya seleccionado para soportar el almacenamiento de los datos.

Diagrama de componentes

El modelo de componentes se debe consignar en el documento de Definición de arquitectura, en la sección de modelo de componentes, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Sección de diseño → Diagrama de componentes.

La diagramación en UML 2.0 se encuentra en la sección [UML](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Diagrama de capas

El modelo de capas se debe consignar en el documento de Definición de arquitectura, en la sección de diagrama de capas, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Sección de diseño → Diagrama de capas.

Diagrama de despliegue

El modelo de despliegue se debe consignar en el documento de Definición de arquitectura, en la sección diagrama de despliegue, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Sección de diseño → Diagrama de despliegue.

El diagrama de despliegue se encuentra en la sección [Diagrama de despliegue](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Diagrama de estado

El diagrama de estados se debe consignar en el documento de Definición de arquitectura, esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto, el documento se debe buscar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Sección de diseño → Diagrama de estados.

El diagrama de estado se encuentra en la sección [Diagrama de Estados](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Catálogo de servicios

La definición del catálogo de servicios y espacio de almacenamiento se puede consultar en la sección de [Catálogo de servicios](#)

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.4.4. Documentos generados en el piloto

Arquitectura Actual

Para visualizar la arquitectura actual se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

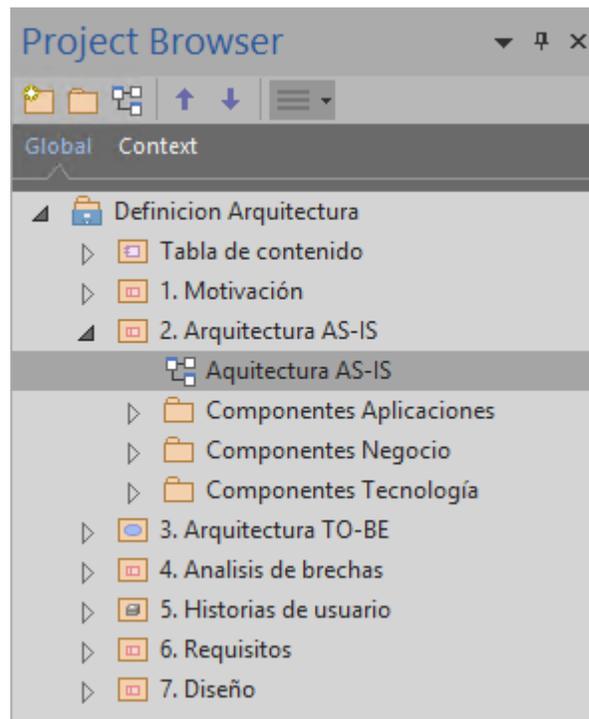


Ilustración 84. Navegación a la arquitectura actual

Arquitectura destino

Para visualizar la arquitectura destino se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

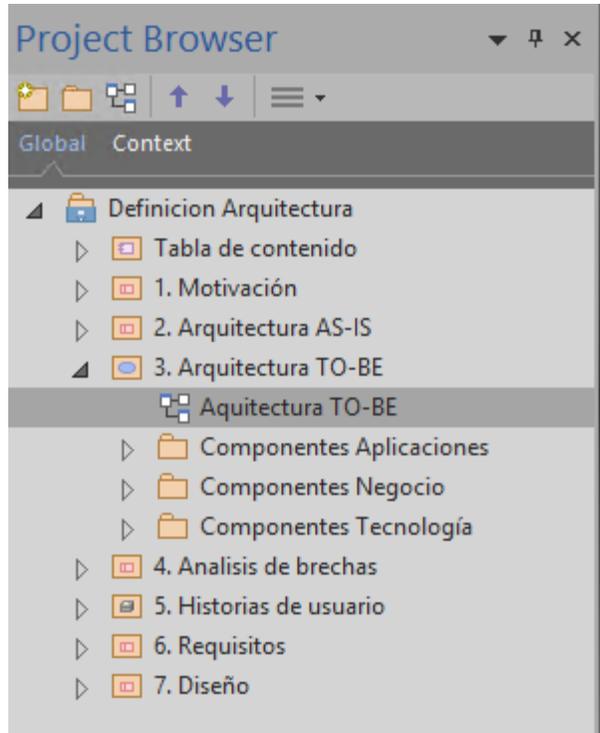


Ilustración 85. Navegación a la arquitectura destino

Brechas

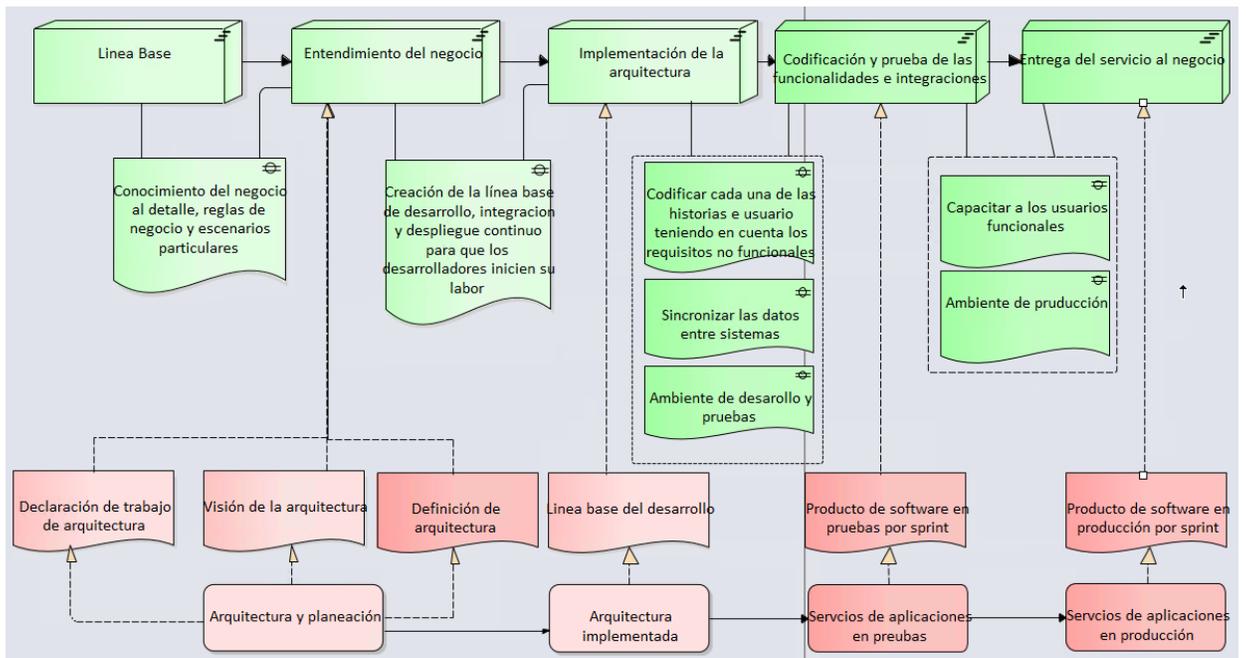


Ilustración 86. Evidencia de las brechas

Diagrama de componentes

Para visualizar el diagrama de componentes se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

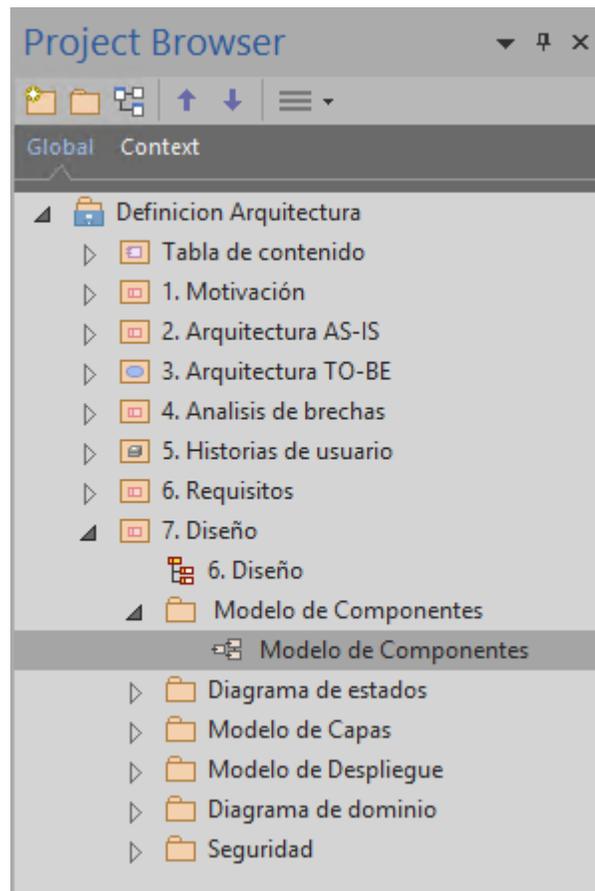


Ilustración 87. Navegación al diagrama de componentes

Diagrama de capas

Para visualizar el diagrama de capas se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

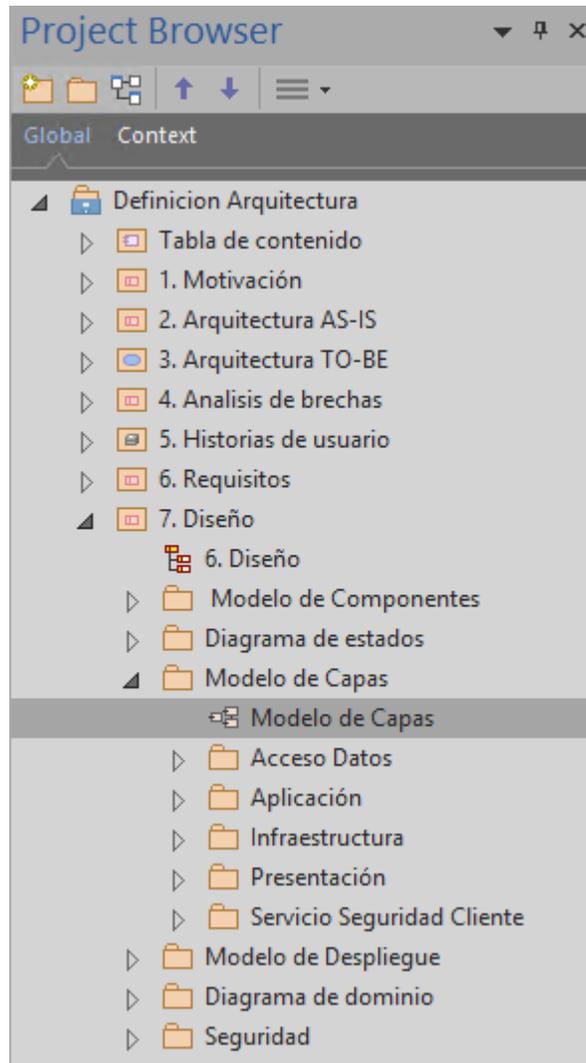


Ilustración 88. Navegación diagrama de capas

Diagrama de despliegue

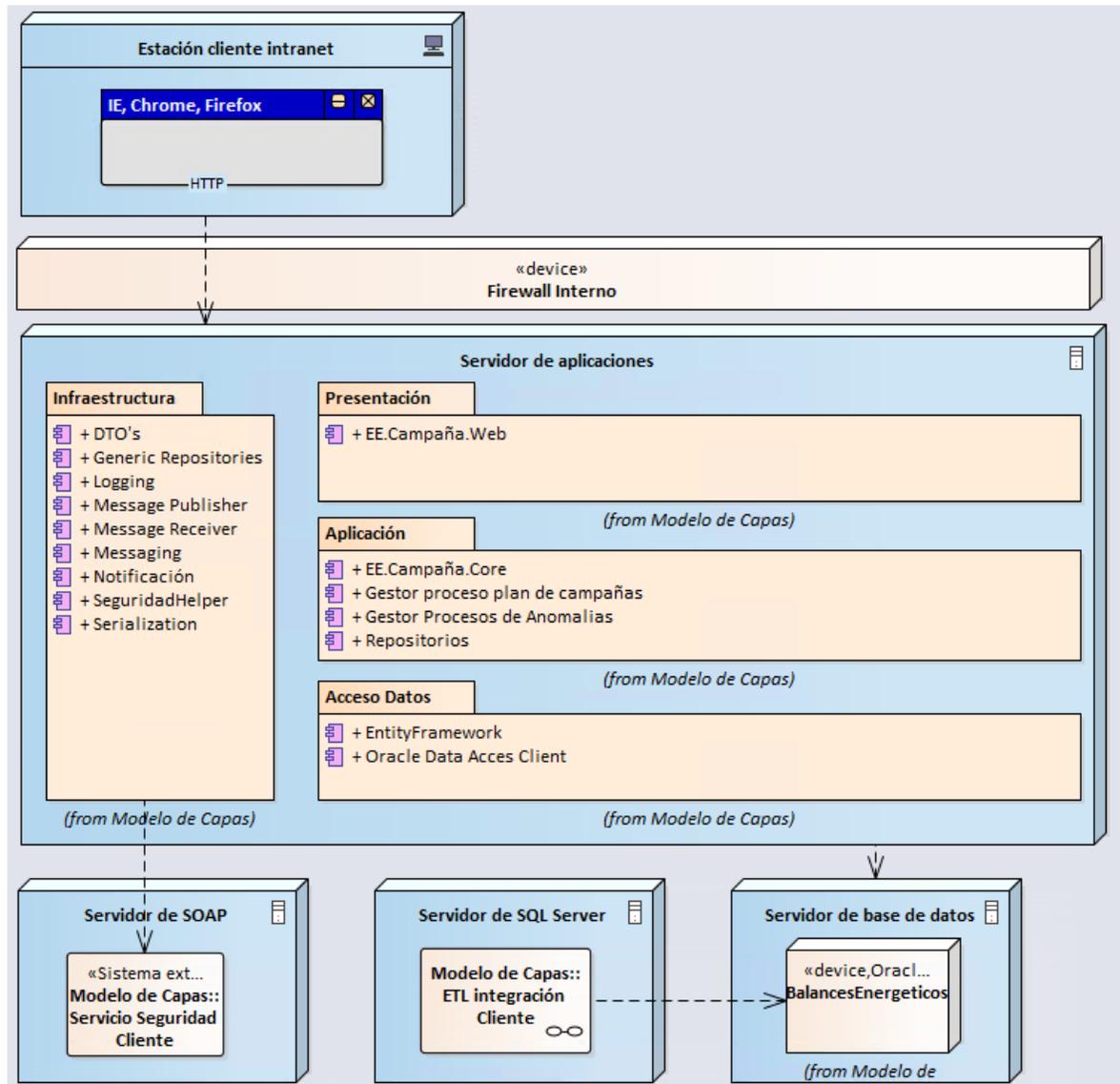


Ilustración 89. Evidencia del diagrama de despliegue

Diagrama de dominio

Para visualizar el diagrama de componentes se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

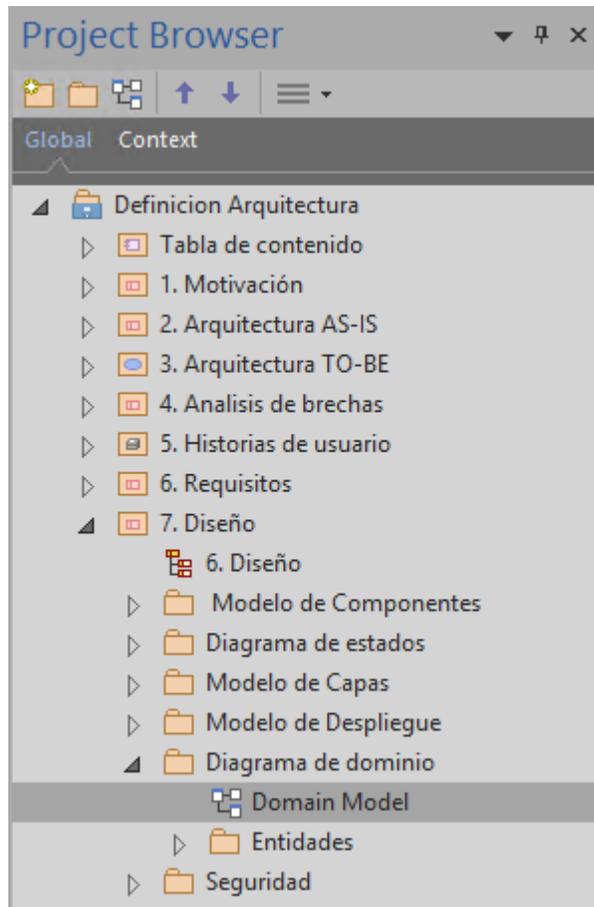


Ilustración 90. Navegación al diagrama de dominio

Diagrama de estados

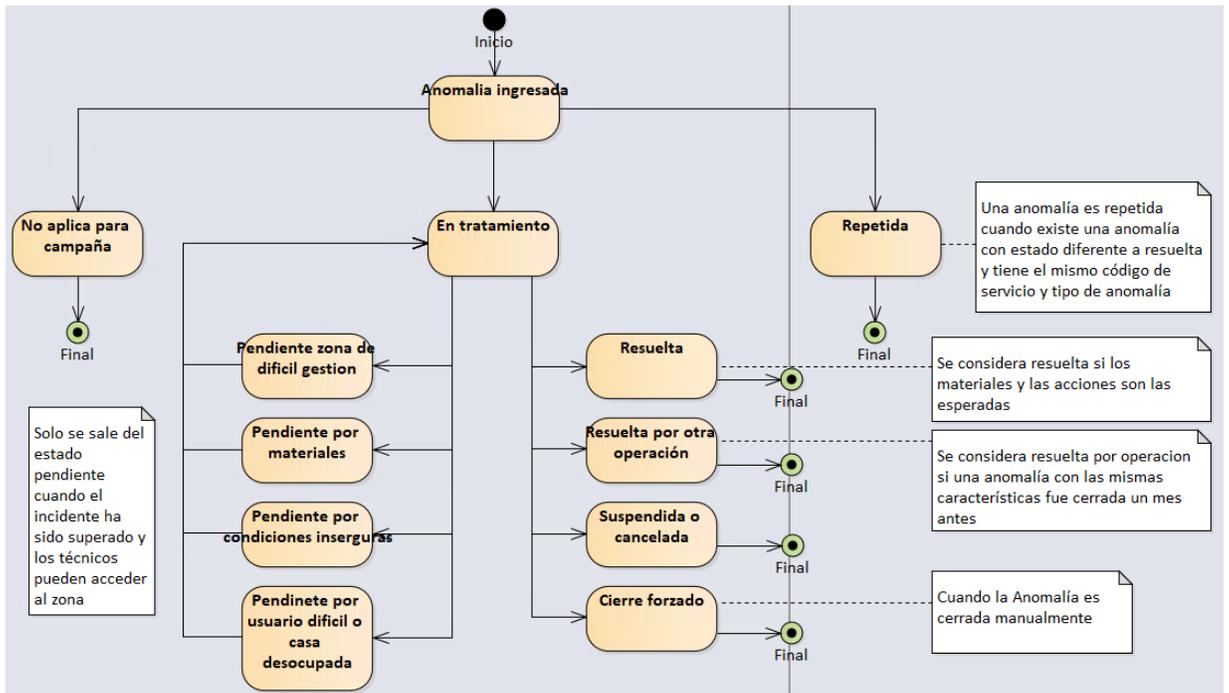


Ilustración 91. Evidencia del diagrama de estados

Prototipos



Ilustración 92. Evidencia de prototipos

6.2.5. Fase D - Arquitectura tecnológica

6.2.5.1. Generalidades de la Fase

The Open Group (2011) resalta que en la arquitectura tecnológica se debe especificar las plataformas tecnológicas y artefactos que soportan todos los servicios que serán consumidos por el sistema de información empresarial que se va a desarrollar.

En esta fase se puede descubrir las plataformas con las cuales operan los sistemas de una organización, información que será vital para el diseño y la arquitectura del sistema de información empresarial que será desarrollado.

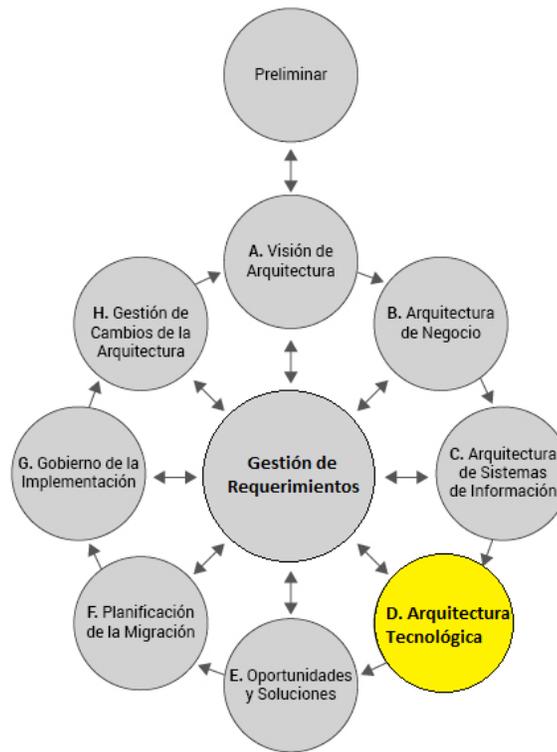


Ilustración 93. Fase D - Arquitectura de Tecnología ⁷⁶

6.2.5.2. Actividades para ejecutar

Entrevistas a los analistas de tecnología y usuarios funcionales

La entrevista con los usuarios funcionales permite la identificación de las plataformas o servicios tecnológicos que tiene implementada la compañía para soportar los sistemas de información que actualmente operan el interior de la compañía, por ejemplo, servidor de base de datos Oracle, versión 10g y 12c, servidor de aplicaciones IIS, entre otros.

⁷⁶ Tomado de (The Open Group, 2011)

El arquitecto debe ser capaz de identificar cuál de estos componentes o plataformas será eliminada o intervenida con el fin de alcanzar la arquitectura de destino. Adicionalmente con esta información también se debe identificar la arquitectura base de tecnología.

Para capturar la data de estas entrevistas se puede usar el formato "Plataformas Tecnologicas.xls" ubicado en el repositorio de arquitectura en la carpeta de plantillas.

Análisis de información

A medida que se toman requisitos y se tiene contacto con los involucrados se identifican nuevos documentos o formatos de apoyo, los cuales deben ser analizados y estudiados por los arquitectos con el fin de identificar información que el usuario no haya expresado en las sesiones de requisitos. Esta es una actividad que se realiza continuamente, con el fin de identificar detalles que los suelen omitir, ya que lo consideran obvio o asumen que es del dominio del arquitecto.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. *(Ver detalles del "Acta" en la sección de artefactos entregados)*

6.2.5.3. Artefactos generados

Arquitectura de línea base de tecnología

La arquitectura base permite la visualización del estado actual de la tecnología de la compañía. Cabe resaltar que solo se diagramarán los componentes tecnológicos que intervienen en los procesos y sistemas de información

afectados por el desarrollo del proyecto. Este diagrama contiene los servidores, características de los mismos, servicios que se prestan, comunicación entre servidores, infraestructura tecnológica, entre otras.

La arquitectura base de la tecnología será desarrollado sobre el archivo "Definición de arquitectura.EAP" en la sección de arquitectura base. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP.

La diagramación de la arquitectura tecnológica se encuentra en la sección [Capa de Tecnología](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Arquitectura destino de la tecnología

La arquitectura de destino permite a los arquitectos diagramar el punto donde se quiere llegar con el desarrollo de la arquitectura. En el repositorio de arquitectura en la carpeta Arquitectura de Referencia se encuentran algunos modelos de referencia que ayudan y facilitan el diseño del sistema de información empresarial. Este diagrama contiene los servidores, características de los mismos, servicios que se prestan, comunicación entre servidores, infraestructura tecnológica, entre otras.

La arquitectura de destino de la tecnología será desarrollada sobre el archivo "Definición de arquitectura.EAP" en la sección de arquitectura destino. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP.

La diagramación de la arquitectura tecnológica se encuentra en la sección [Capa de Tecnología](#). En esta sección se explican cada uno de los componentes y la relación que existe entre cada uno de ellos.

Brechas

La definición de brechas y espacio de almacenamiento se puede consultar en la sección de [Brechas](#)

Catálogo de servicios

La definición del catálogo de servicios y espacio de almacenamiento se puede consultar en la sección de [Catálogo de servicios](#)

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.6. Fase E - Oportunidades y soluciones

6.2.6.1. Generalidades de la fase

The Open Group (2011) define la fase de oportunidades y soluciones como la primera fase de la implementación, ya que se describen los procesos para entregar las arquitecturas de destino identificadas en fases anteriores. Adicionalmente afirma que se debe tener en cuenta las brechas identificadas entre la arquitectura base y la arquitectura destino de cada dominio.

El entregable principal de esta fase es el mapa de ruta donde se identifican y priorizan cada uno de los módulos y componentes que se deben desarrollar. Adicionalmente se identifica si estos componentes serán desarrollados como parte de un proyecto, programa o cartera. Cabe resaltar que para definir el mapa de ruta es importante conocer las capacidades y restricciones tecnológicas y del equipo de trabajo.

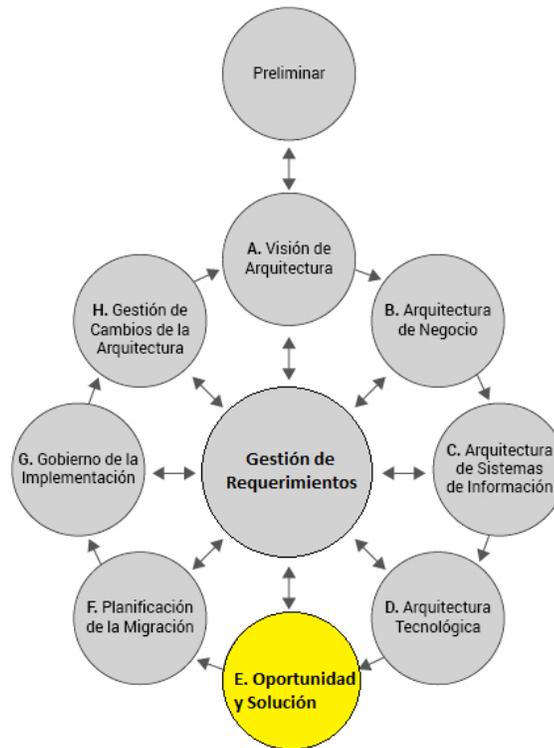


Ilustración 94. Fase E - Oportunidades y Soluciones ⁷⁷

6.2.6.2. Actividades para ejecutar

Identificación de limitaciones

Teniendo en cuenta los componente y módulos a desarrollar es importante identificar las limitaciones o restricciones con que cuenta la compañía y el equipo de desarrollo, de esta forma será más consciente la toma de decisiones, por ejemplo: aceptar tiempos más elevados en el desarrollo, contratar más personal o adquisición de nuevos recursos. *(Ver detalles de las “Limitaciones” en la sección de artefactos entregados)*

Reunión de priorización de componentes y módulos

En esta reunión debe estar conformada por los directivos y el equipo de desarrollo del proyecto, con el fin de exponer los componentes y módulos a desarrollar, cada ítem debe tener una pequeña explicación, objetivo

⁷⁷ Tomado de (The Open Group, 2011)

estratégico al que apunta, prioridad y tiempo estimado de desarrollo. Luego de que todo el equipo conozca la información mencionada anteriormente se inicia una discusión con el fin de conocer sus expectativas y hacer modificaciones a la propuesta si esta lo requiere.

Cabe resaltar que se deben tener en cuenta las limitaciones y capacidades antes de comprometerse con la organización, es decir, si el desarrollo de un proyecto lo necesitan para el siguiente mes y no cuentan con el personal necesario, este es el espacio para discutirlo y buscar soluciones. *(Ver detalles del “Mapa de ruta” en la sección de artefactos entregados)*

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. *(Ver detalles del “Acta” en la sección de artefactos entregados)*

Entrega de proyecto para su ejecución

Luego de la aprobación del mapa de ruta, se entrega formalmente los proyectos a desarrollar a la oficina de proyectos de la empresa, con el fin de que sean ellos quienes lideren del desarrollo y la materialización de los mismo, teniendo en cuenta los lineamientos, restricciones y capacidades identificadas en los pasos anteriores.

6.2.6.3. Artefactos generados

Mapa de ruta

The Open Group (2011) define el mapa de ruta de la arquitectura como una lista de componentes y módulos en una línea de tiempo. Cada componente o

módulo identifica un grupo lógico de cambios o requerimientos necesarios para realizar la Arquitectura de destino.

En el marco de trabajo adaptado los mapas de ruta se desarrollarán según la clasificación de su desarrollo, es decir, pueden ser un proyecto o hacer parte de un programa o cartera. Con lo anterior se está buscando que todos los sistemas de información empresariales que se desarrollen al interior de una compañía tengan una gestión integral y no se repitan esfuerzos. Cabe resaltar que esta es una primera versión del mapa de ruta, ya que en la fase de planificación de la migración será afinada.

Los mapas de ruta serán almacenados en el repositorio de arquitectura, en la carpeta Mapas de ruta. Estos archivos no se colocarán al interior de cada carpeta, ya que como se expresó anteriormente en mucha ocasión un mapa de ruta está compuesto de uno o varios componentes o módulos de diversos sistemas de información empresarial. La plantilla para su generación inicial se encuentra en el repositorio de arquitectura en la carpeta de plantillas, bajo el nombre de "RoadMap.xlsx"

Limitaciones

Las limitaciones serán almacenadas en el archivo "*Definición de arquitectura.EAP*" en la sección de motivación. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Definición de arquitectura.EAP → Motivación.

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.6.4. Documentos generados en el piloto

Mapa de Ruta



Objetivo Estratégico	Proyecto	Entregable	2017							
			FEB	ABR	JUL	AGO	SEP	OCT	NOV	DIC
Disminuir un 80% la pérdida de energía Aumentar los Ingresos en un 30% Aumentar unicidad, consistencia, completitud, oportunidad y confidencialidad en los activos en información	Desarrollo del sistema de información para la gestión de balances	Balance de estructuras								
		Balance de Macromediciones								
		Balance de circuitos								
		Balance de 34,5								
		Balance de 13,2								
		Balance de distribución								
		Balance Unifilar								
	Desarrollo del sistema de información para la gestión de campañas	Maestros de la aplicación	Entrega Total							
		Gestión de anomalías		Entrega Total						
		Gestión de campañas			Entrega Total					
		Generación de Informes		Entrega Parcial	Entrega Total					
		Integración con otros sistemas		Entrega Parcial	Entrega Total					

Ilustración 95. Evidencia de mapas de ruta

6.2.7. Fase F - Planificación de la migración

6.2.7.1. Generalidades de la fase

The Open Group (2011) afirma que la fase de planificación de la migración aborda la estratégica de cómo llegar desde la arquitectura base a la arquitectura destino, generando un plan de implementación y migración. Cabe resaltar que el plan de implementación es un trabajo que se debe realizar en compañía de la oficina de proyectos o entidad encargada de ejecutar los proyectos al interior de una compañía, ya que son ellos los que conocen los lineamientos, políticas y gobierno del ciclo de vida de los proyectos.

Adicionalmente, en esta etapa el mapa de ruta se afina teniendo en cuenta la evolución de costos, dependencias y beneficios que el desarrollo de los proyectos entrega a la compañía.

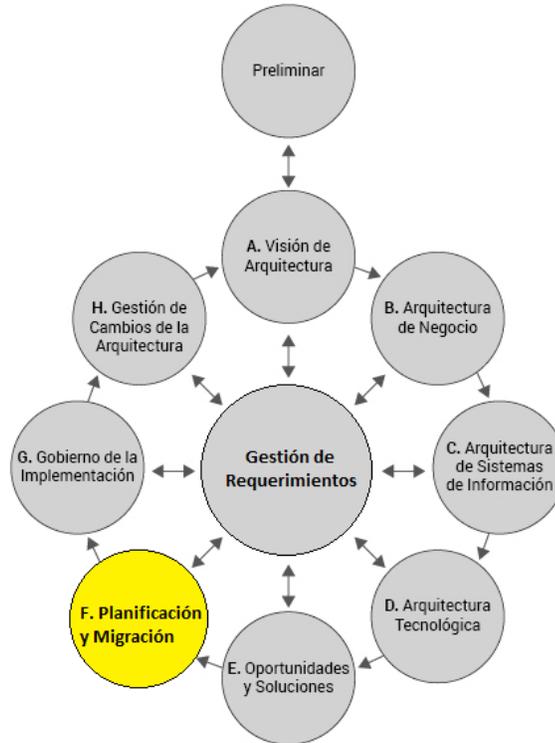


Ilustración 96. Fase F - Planeación de la migración ⁷⁸

6.2.7.2. Actividades para ejecutar

Estimación de recursos y cronograma

La estimación de recursos y el desarrollo del cronograma es una actividad que se debe realizar en conjunto con la oficina de proyectos, ya que esta área es la encargada de ejecutar los proyectos y adquirir los recursos necesarios para su desarrollo. Es en este punto donde se interactúa con otras metodologías o marcos de trabajo adaptados por la empresa. El objetivo principal es trabajar conjuntamente con las otras áreas y no repetir esfuerzos o generar conflictos entre áreas y procesos.

⁷⁸ Tomado de (The Open Group, 2011)

Para la elaboración del cronograma es de gran importancia tener como base el *product backlog*, ya que este contiene el alcance del desarrollo del sistema de información empresarial, la estimación de la construcción de la aplicación se realizará usando la metodología de puntos por cada historia de usuario identificada en la toma de requisitos. Adicionalmente se debe tener en cuenta el análisis de capacidades y restricciones realizado en fases anteriores, con el fin de generar una planeación realista acorde a las capacidades de la empresa y alcance del proyecto.

Los artefactos generados en esta actividad, será los definidos por la metodología de gestión de proyectos de la compañía. *(Ver detalles del “Plan de implementación y migración” en la sección de artefactos entregados)*

Socialización del documento de arquitectura y mapa de ruta

En esta etapa del desarrollo de la arquitectura empresarial, el documento de arquitectura y la línea base ya deben estar desarrolladas y afinadas, por lo que es el momento de socializar a todos los interesados el resultado del mismo, adicionalmente se debe presentar la planeación realizada en conjunto con la oficina de proyectos.

Si llegase a salir modificaciones, sugerencias o cambios estas definiciones se deben tratar como un control de cambios a la arquitectura desarrollada. La gestión del cambio se amplía en la sección 6.2.9 del documento.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. *(Ver detalles del “Acta” en la sección de artefactos entregados)*

6.2.7.3. Artefactos generados

Plan de implementación y migración

Josey (2013) define el plan de implementación y migración como una guía que proporciona la estrategia y cronograma de los proyectos requeridos para la implementación de la arquitectura destino.

Para generar el plan de implementación y migración se debe usar la plantilla “*Plan de implementación y migración.doc*” que se encuentra en el repositorio de arquitectura al interior de la carpeta plantillas. Cuando el documento haya sido diligenciado se debe almacenar en el repositorio de arquitectura en la carpeta Arquitectura → Arquitectura a trabajar → Documentos de gestión.

Mapa de ruta

La definición mapas de ruta y espacio de almacenamiento se puede consultar en la sección de [Mapa de Ruta](#)

Definición de arquitectura

El documento de arquitectura almacena la arquitectura base y de destino del negocio, aplicaciones, datos e infraestructura, adicionalmente deben estar las historias de usuario y requisitos identificados en los talleres y sesiones de trabajo realizados en fases anteriores. Cabe resaltar que solo se diagramarán los componentes tecnológicos que intervienen en los procesos y sistemas de información afectados por el desarrollo de la arquitectura.

EL documento de arquitectura será desarrollado sobre el archivo “Definición de arquitectura.EAP”. Esta plantilla se encuentra en el repositorio de arquitectura en la carpeta plantillas, y si ya ha sido convertido en artefacto el documento se encuentra en el repositorio de arquitectura en la carpeta proyecto → Proyecto a trabajar → Definición de arquitectura.EAP.

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.7.4. Documentos generados en el piloto

Plan de Implementación y migración



Nombre ↑	Propietario	Última modificación	Tamaño del archivo
W Declaracion del trabajo de arquitectura.docx	yo	19 feb. 2018 yo	124 KB
W Plan Implementacion y Migracion.docx	yo	3 mar. 2018 yo	88 KB
W Visión de arquitectura.docx	yo	19 feb. 2018 yo	79 KB

Ilustración 97. Evidencia del plan de implementación y migración

6.2.8. Fase G - Gobierno de la implementación

6.2.8.1. Generalidades de la Fase

The Open Group (2011) afirma que en la fase H – Gobierno de implementación es el desarrollo y la materialización de los paquetes de trabajo planeados en el mapa de ruta, adicionalmente sugiere que se hagan entregas continuas con el fin de que los interesados perciban el valor de los desarrollos.

En el marco de trabajo adaptado los desarrollos de los sistemas de información empresarial se realizarán con la metodología Scrum, donde se iniciará con una planeación de los paquetes de trabajo con mayor prioridad (los cuales son establecidos por el mapa de ruta), diariamente se realizarán seguimientos con el fin de atender errores y problemas de forma temprana y finalmente entregar a los interesados un producto funcional.

La metodología Scrum se encuentra en la sección [Scrum](#). En esta sección se explica la metodología, los entregables, las ceremonias y los actores que intervienen en Scrum.

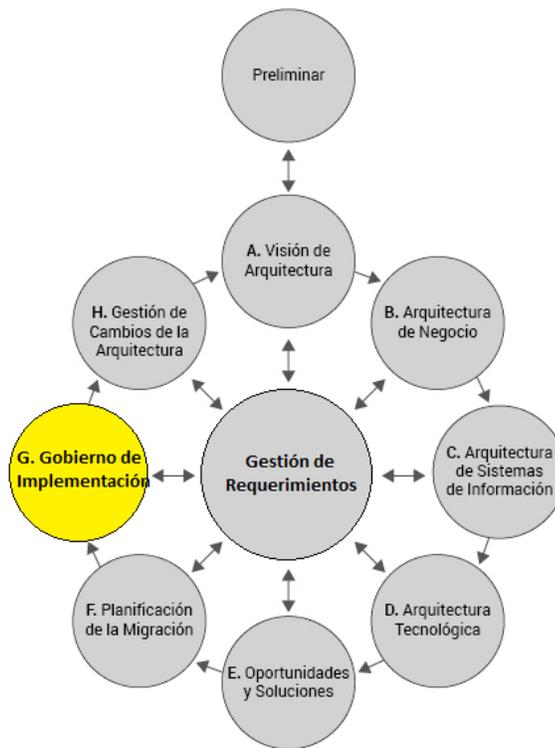


Ilustración 98. Fase G - Gobierno de implementación ⁷⁹

6.2.8.2. Actividades para ejecutar

Auditorias

Las auditorias son usadas para garantizar el cumplimiento de los acuerdos de arquitectura realizados entre los colaboradores y patrocinadores, con el fin de:

- Garantizarla calidad de los productos y el cumplimiento de los compromisos.

⁷⁹ Tomado de (The Open Group, 2011)

- Adhesión de principios y requerimientos de la arquitectura que se está desarrollando.
- Identificación de reprocesos y prevención de riesgos de forma temprana
- Garantizar el cumplimiento de los procesos

(Ver detalles de la “Evaluación de conformidad” en la sección de artefactos entregados)

Reuniones de planeación

La ceremonia de planeación consiste en la identificación del alcance que será desarrollado en el sprint. Debe estar todo el equipo del proyecto presente ya que cada uno conoce su capacidad de desarrollo y de esta forma comprometerse con el incremento del sistema de información empresarial. La planeación está compuesta de dos actividades macro, la primera es la creación del sprint backlog, que es conocido como una porción del *product backlog* y la segunda es la definición de cómo se van a desarrollar el *sprint backlog*.

Esta actividad genera el sprint Backlog, el cual está compuesta por la lista de historias de usuario que se deben desarrollar durante el sprint planeado.

Reuniones diarias de seguimiento

La reunión diaria es una actividad de quince minutos, donde cada integrante del equipo de desarrollo informa en qué avanzó el día anterior, con qué se compromete para el día actual y si ha tenido algún inconveniente o problema. Si el problema no es muy extenso se discuten soluciones con el resto del equipo de desarrollo, de esta forma se aprovecha el conocimiento de los compañeros para solucionar incidentes o encontrar alternativas.

Reuniones de retrospectiva

La retrospectiva del sprint es una reunión del equipo Scrum donde cada integrante hace un aporte de mejora y/o resalta las buenas prácticas ejecutadas en el proyecto, es decir, es un espacio de lecciones aprendidas,

que permite al equipo madurar, crecer y cohesionarse con el fin de mejorar desde lo técnico hasta las habilidades blandas.

Reuniones de entrega

La revisión del sprint es la entrega formal del incremento del sistema de información empresarial a todos los involucrados, incluyendo los desarrolladores, usuarios funcionales, directivos y líderes del proyecto. El objetivo es hacer un pequeño demo o presentación de las nuevas funcionalidades, las cuales están disponibles para su uso inmediato.

Aprobación

Es de alta importancia que los artefactos generados sean socializados y aprobados por los interesados, de esta forma se pueden continuar con las siguientes fases. Cabe resaltar que es un proceso iterativo, por lo que es posible devolverse a hacer modificaciones a documentos o arquitecturas que ya han sido revisadas, sin embargo, lo importante es que cada cambio esté revisado y aprobado. Las aprobaciones deben quedar soportadas en un acta firmada por el responsable de la aprobación. *(Ver detalles del “Acta” en la sección de artefactos entregados)*

Gestión de errores reportados

La gestión de errores permite una correcta solución de los incidentes y generar lecciones aprendidas con el fin de retroalimentar al equipo de desarrollo. Esta gestión debe garantizar el fácil reporte de los errores por parte de los funcionales, la asignación al desarrollador y la satisfactoria solución de los mismos. Adicionalmente, la comunicación debe ser vital, con el fin de que todos los interesados conozcan el estado de la solución del incidente y se genere tranquilidad sobre la calidad del sistema de información empresarial que se está desarrollando

Cabe resaltar que los errores deben ser reportados en una herramienta que permita hacer seguimiento y gestión a los incidentes, esta herramienta o metodología debe ser identificada en el plan de Implementación y Migración.

Algunas herramientas que la compañía puede usar para la gestión son las siguientes: Mantis, *WorkFront*, RT (*Request Tracker*), aranda, entre otros, cabe resaltar que existen algunas GNU y otras licenciadas.

Cabe resaltar que este proceso se debe adaptar a las políticas o procedimientos que tiene la mesa de servicio de la compañía.

6.2.8.3. Artefactos generados

Solicitudes del cambio

Las solicitudes de cambio son modificaciones y cambios de alcance al proyecto identificadas a partir de la fase F – planificación de la migración. Esta petición debe ir por escrito, con el fin de ser evaluadas por el comité de arquitectura. Las solicitudes de cambio deben ser muy explícitas, con el fin de no generar ambigüedad o el no entendimiento de la solicitud por parte del comité de arquitectura. Adicionalmente debe tener los impactos del cambio en cada una de las arquitecturas.

El formato de solicitud de cambio se encuentra en el repositorio de arquitectura en la carpeta plantillas, bajo el nombre “Control de Cambios.docx”. Cuando este formato haya sido diligenciado se debe almacenar en el repositorio de arquitectura en la siguiente ruta de carpetas: Arquitecturas → Arquitectura a trabajar → Controles de cambio

Paquete de trabajo o incremento

El paquete de trabajo o incremento son las funcionalidades desarrolladas y probadas por el equipo de desarrollo, deben ser puestas en ambiente de pruebas, con el fin de que los usuarios funcionales inicien sus pruebas y puedan validar que el paquete de trabajo o incremento está acorde a las necesidades y solicitudes del proceso.

En caso de que sean encontrados errores, estos deben ser reportados por los funcionales al equipo de desarrollo usando una herramienta de gestión de incidentes, con el fin de hacer una correcta gestión de los mismas, garantizar

su corrección y enriquecer la retroalimentación del trabajo realizado por el equipo.

Sprint Backlog

El *sprint Backlog* consiste en las historias de usuario seleccionadas para desarrollar en el sprint planificado. Cada una de las historias de usuario deben quedar detalladas en actividades, con el fin de hacer un seguimiento al avance y cumplimiento. Existen varias herramientas que soportan la creación de los *sprint Backlog*, en caso de que una empresa cuente con dichas herramientas pueden ser usadas, sin embargo, en el marco de trabajo adaptado se entregará un formato con un cronograma donde se identifiquen las actividades que hacen parte de una historia de usuario, el tiempo y esfuerzo que requiere su desarrollo

El formato de solicitud de cambio se encuentra en el repositorio de arquitectura en la carpeta plantillas, bajo el nombre “Sprint backlog_XX.xls”. Cuando este formato haya sido diligenciado se debe almacenar en el repositorio de arquitectura en la siguiente ruta de carpetas: Arquitecturas → Arquitectura a trabajar → Sprint Backlog.

Evaluación de conformidad

La evaluación de conformidad es el resultado que se genera luego de la ejecución de una auditoría, dónde se revisa el cumplimiento de aspectos como: cumplimiento con el marco de trabajo adaptado, cumplimiento con el diseño arquitectónico, cumplimiento con el desarrollo del sistema de información empresarial.

El formato de evaluación de conformidad se encuentra en el repositorio de arquitectura en la carpeta plantillas, bajo el nombre de “Evaluación de conformidad_AAAAMMDD.xls”. Cuando este formato haya sido diligenciado se debe almacenar en el repositorio de arquitectura en la siguiente ruta: Arquitecturas → Arquitectura a trabajar → Auditorías

Acta

La definición del acta y espacio de almacenamiento se puede consultar en la sección de [Acta](#)

6.2.8.4. Documentos generados en el piloto

Auditorias

Mi u... > Trabajo... > Entre... > Repositorio de arq... > Arquite... > Arquitectura Ca... > Auditori... ▾ 👤 ☰

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
 Evaluación de Conformidad - Arquitecturas.xlsx 👤	yo	16:56 yo	58 KB
 Evaluación de Conformidad - Marco de trabajo.xlsx 👤	yo	16:56 yo	57 KB
 Evaluación de Conformidad - Requisitos.xlsx 👤	yo	16:56 yo	58 KB

Ilustración 99. Evidencia de las auditorias

Spring Backlog

Mi u... > Trabajo... > Entre... > Repositorio de ar... > Arquite... > Arquitectura C... > Sprint Backl... ▾ 👤 ☰ (

Nombre ↑	Propietario	Última modificación	Tamaño del archivo
 Sprint Backlog_1.xlsx 👤	yo	16:56 yo	46 KB
 Sprint Backlog_2.xlsx 👤	yo	16:56 yo	48 KB
 Sprint Backlog_3.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_4.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_5.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_6.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_7.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_8.xlsx 👤	yo	16:56 yo	47 KB
 Sprint Backlog_9.xlsx 👤	yo	16:56 yo	47 KB

Ilustración 100. Evidencia de planeación de sprint

6.2.9. Fase H - Gestión del cambio de la arquitectura

6.2.9.1. Generalidades de la Fase

The Open Group (2011) afirma que el proceso de gestión de cambio de la arquitectura busca garantizar que la arquitectura no pierda el valor de negocio con los cambios solicitados por los interesados en el desarrollo de todo el marco de trabajo. Busca asegurar que los cambios a la arquitectura sean coherentes y alineados con los objetivos estratégicos.

Las solicitudes de cambio serán evaluadas por el comité de arquitectura definido por la compañía, en este comité se identificará la magnitud del cambio y las arquitecturas o paquetes de trabajo que serán impactados en caso de aprobarse su desarrollo, por ejemplo: Arquitectura de negocio, arquitectura de aplicaciones, arquitectura de datos, arquitectura tecnológica, requerimientos, mapa de ruta, planes de trabajo, entre otros.

Cabe resaltar que es de gran importancia que todos los artefactos impactados por el cambio se les genere una nueva versión, con el fin de llevar una bitácora y permitir el análisis de la trazabilidad en el desarrollo de la arquitectura.

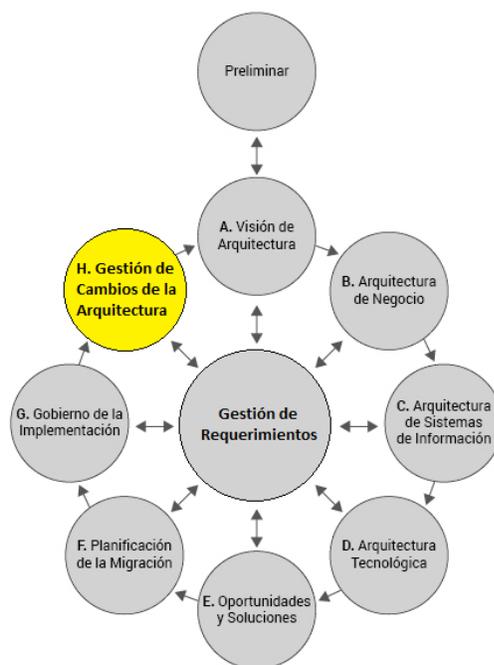


Ilustración 101. Fase H - Gestión del Cambio de Arquitectura⁸⁰

⁸⁰ Tomado de (The Open Group, 2011)

6.2.9.2. Actividades para ejecutar

Identificar los riesgos

Esta actividad permite analizar e identificar riesgos adicionales a los escritos por el solicitante en el documento de control de cambios. El comité de cambios debe analizar en conjunto con los interesados que solicitaron el cambio qué otros riesgos existen adicionales a los identificados inicialmente.

Comité de arquitectura

El comité de arquitectura es el encargado de analizar detalladamente el cambio, en este comité se analiza el documento de control de cambios entregado por el solicitante, sin embargo, se entran en discusiones para identificar de qué otra forma son afectadas las dimensiones arquitectónicas y en qué otros riesgos se incurren en su desarrollo. Cabe resaltar que es de gran importancia que el cambio solicitado no vaya en contra de los objetivos estratégicos los cuales se buscan cumplir con el desarrollo de la arquitectura.

Documentar los cambios en el documento de arquitectura

Se deben documentar los cambios aprobados en el comité de arquitectura en el documento de arquitectura. Cabe resaltar que es necesario generar una nueva versión del documento antes de realizar las modificaciones con el fin de conservar las líneas bases que habían sido definidas y conocer el origen del cambio.

Las modificaciones al documento de arquitectura incluyen las capas de negocio, datos, sistemas de información, tecnología, motivaciones, implementación y migración y requisitos.

6.2.9.3. Artefactos generados

Definición de Arquitectura

La definición de la arquitectura y espacio de almacenamiento se puede consultar en la sección de [Definición de Arquitectura](#)

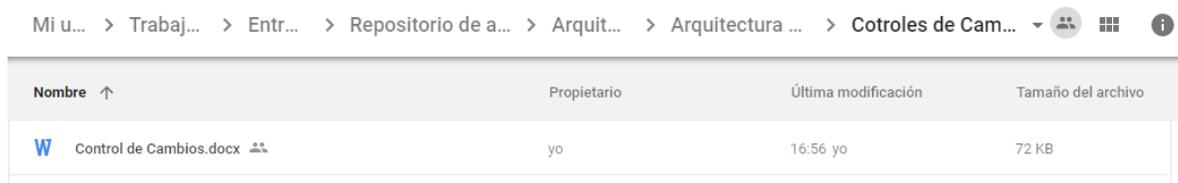
Respuesta a la solicitud del cambio (Acta)

El resultado final del comité es una respuesta formal del cambio, es decir, si aprueba o se rechaza. Esta respuesta se realiza a través de un acta donde queda consignadas la actividades, observaciones, sugerencias y respuestas relacionadas con la petición de control de cambios.

El formato del acta se encuentra en el repositorio de arquitectura en la carpeta de plantillas, bajo el nombre de “Acta de Reunión.xls”. Cuando este formato haya sido diligenciado y firmado se debe almacenar en repositorio de arquitectura en la siguiente ruta de carpetas: Arquitectura → Arquitectura a trabajar → Actas.

6.2.9.4. Documentos generados en el piloto

Control de Cambios



Nombre ↑	Propietario	Última modificación	Tamaño del archivo
 Control de Cambios.docx 	yo	16:56 yo	72 KB

Ilustración 102. Evidencia de los controles de cambio

6.2.10. Gestión de requerimientos

6.2.10.1. Generalidades de la fase

The Open Group (2011) afirma que la gestión de requisitos es un proceso dinámico en el cual los requisitos se completan y se afinan a medida que se van ejecutando las diferentes fases. Se resalta que en esta actividad es muy importante la identificación de conductores, limitaciones, restricciones y capacidades que tiene la compañía, con el fin de garantizar que los requisitos son viables en el desarrollo de la arquitectura empresarial.

Este proceso solo se encarga de la captura de requisitos, sin embargo, la priorización y la eliminación de los mismos se realiza en las fases E (Oportunidades y soluciones) y la fase F (Planificación de la migración)

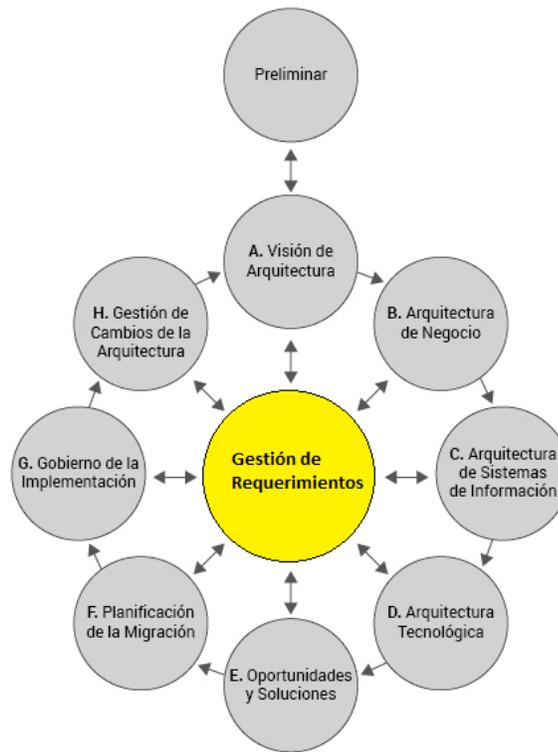


Ilustración 103. Gestión de Requerimientos ⁸¹

6.2.10.2. Actividades para ejecutar

La gestión de requisitos se realizará a través de diferentes actividades y talleres según la necesidad de la fase donde se ejecuten. La siguiente lista muestra las técnicas usadas en el marco de trabajo adaptado.

- Historias de usuario ([Historias de Usuario](#))
- Entrevistas ([Entrevistas](#))
- Análisis de documentos ([Análisis de documentos](#))
- *Design Thinking* ([Design Thinking](#))
- QAW ([QAW](#))

⁸¹ Tomado de (The Open Group, 2011)

Las técnicas de elicitación de requisitos se encuentran en la sección [Elicitación de requisitos](#).

Cabe resaltar que en cada una de las fases se hace énfasis en la técnica que se debe utilizar, con el fin de orientar a los arquitectos la forma más adecuada de lograr los objetivos, según las necesidades de cada fase.

Auditorias

Las auditorias son usadas para garantizar el cumplimiento de los acuerdos de arquitectura realizados entre los colaboradores y patrocinadores, con el fin de:

- Garantizarla calidad de los productos y el cumplimiento de los compromisos.
- Adhesión de principios y requerimientos de la arquitectura que se está desarrollando.
- Identificación de reprocesos y prevención de riesgos de forma temprana
- Garantizar el cumplimiento de los procesos

(Ver detalles de la “Evaluación de conformidad” en la sección de artefactos entregados)

6.2.10.3. Artefactos generados

Requisitos funcionales

La definición de los requisitos funcionales y espacio de almacenamiento se puede consultar en la sección de [Requisitos Funcionales](#)

Requisitos no funcionales

La definición de los requisitos no funcionales y espacio de almacenamiento se puede consultar en la sección de [Requisitos no Funcionales](#)

Modelo de dominio

La definición del modelo de dominio y espacio de almacenamiento se puede consultar en la sección de [Modelo de dominio](#)

Evaluación de conformidad

La definición de evaluación de conformidad y espacio de almacenamiento se puede consultar en la sección de [Evaluación de Conformidad](#)

Definición de Arquitectura

La definición de la arquitectura y espacio de almacenamiento se puede consultar en la sección de [Definición de Arquitectura](#)

6.2.10.4. Documentos generados en el piloto

Requerimientos

Para visualizar los requerimientos se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas -> Arquitectura Campañas -> Definición de arquitectura. Si tiene la aplicación de Enterprise Architect abrir el archivo Definición de Arquitectura.EAP, en caso contrario ingresar a la carpeta Definición de Arquitectura y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

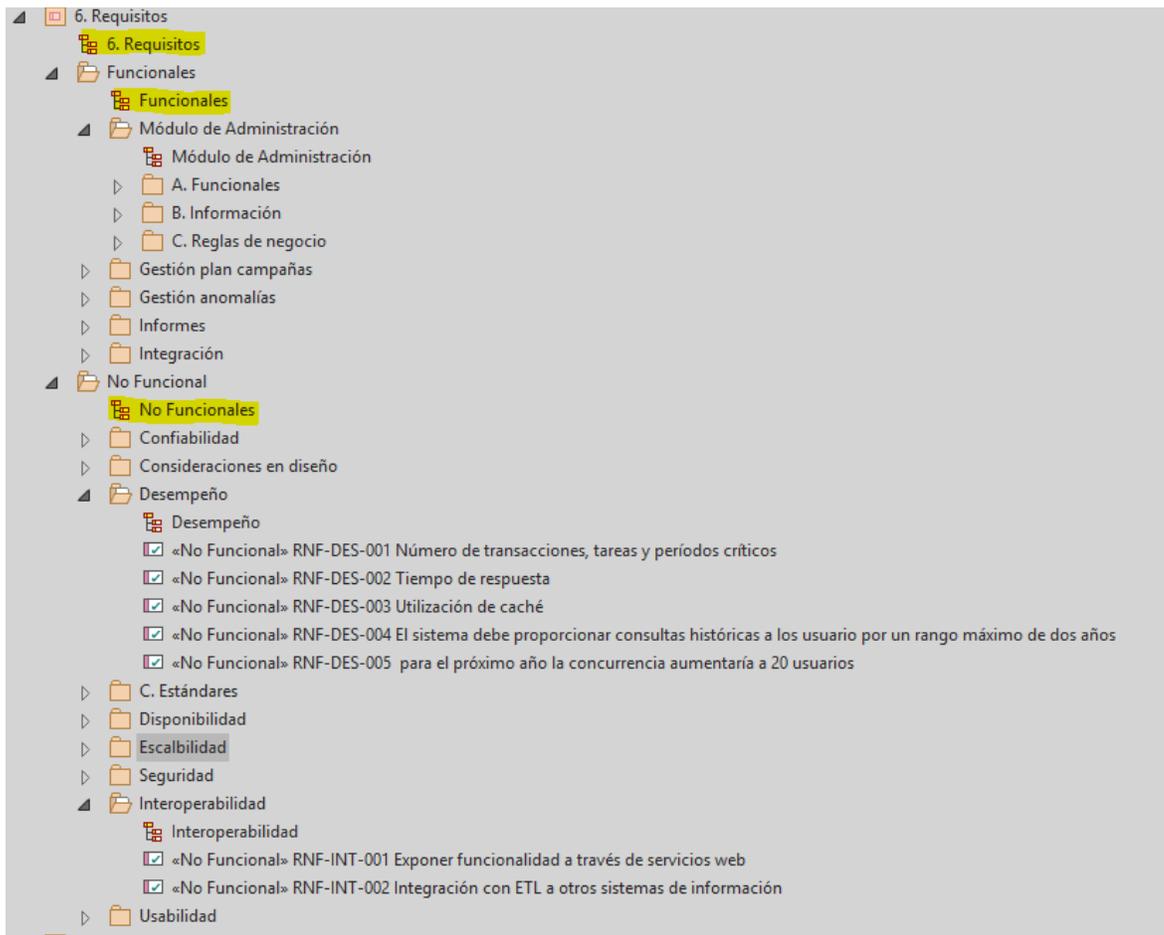


Ilustración 104. Navegación a los requisitos funcionales y no funcionales

6.2.11. Arquitecturas de referencia

Teniendo en cuenta las afirmaciones de Singh P.M. (2017) sobre el uso de la arquitectura de referencia se entregan algunas, con el fin de orientar y ayudar a las empresas y arquitectos a tomar mejores decisiones en el diseño de solución.

Se van a tener unas imágenes de referencia para el trabajo escrito, sin embargo, del detalle y las observaciones de las arquitecturas se encuentran en el repositorio de arquitectura en la carpeta de arquitecturas de referencia, en el archivo nombrado “Arquitecturas de referencia.EAP”.

6.2.11.1. Arquitectura de referencia para aplicaciones web

Diagrama de capas

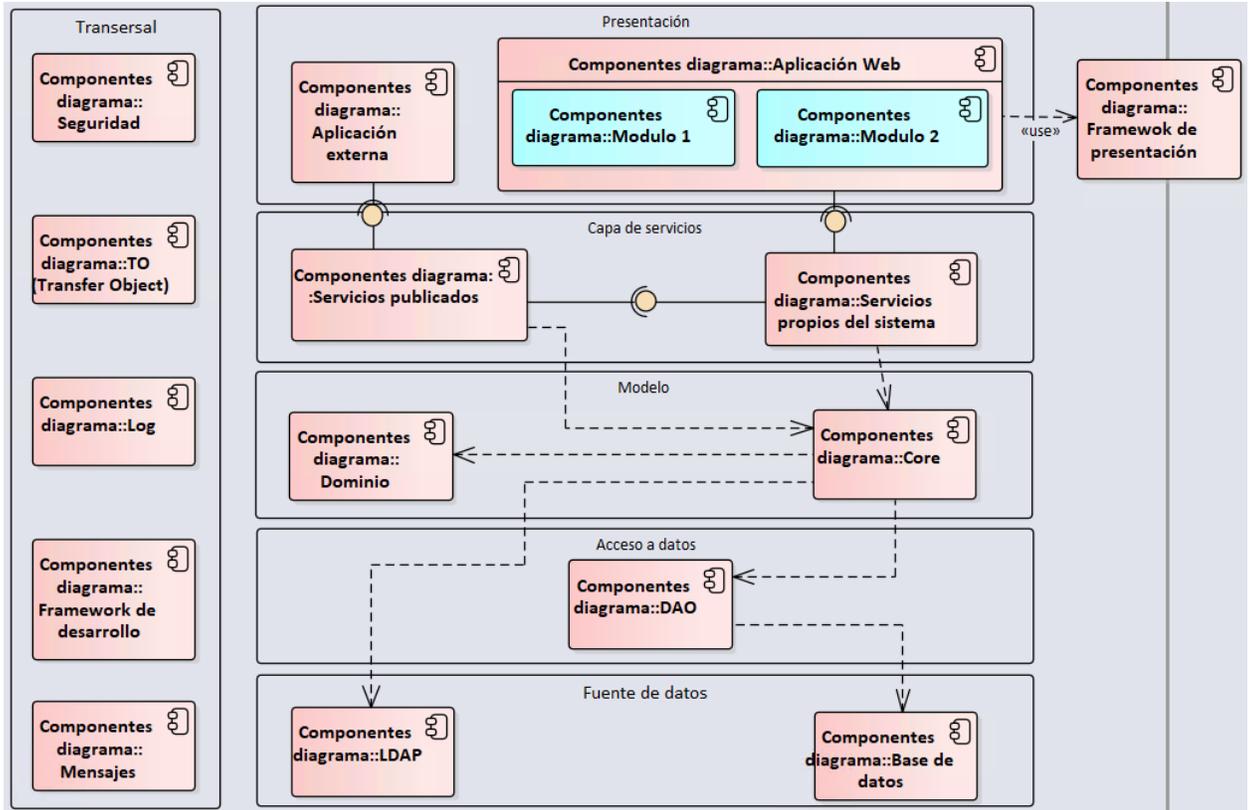


Ilustración 105. Arquitectura de aplicaciones web, diagrama de capas

Diagrama de distribución

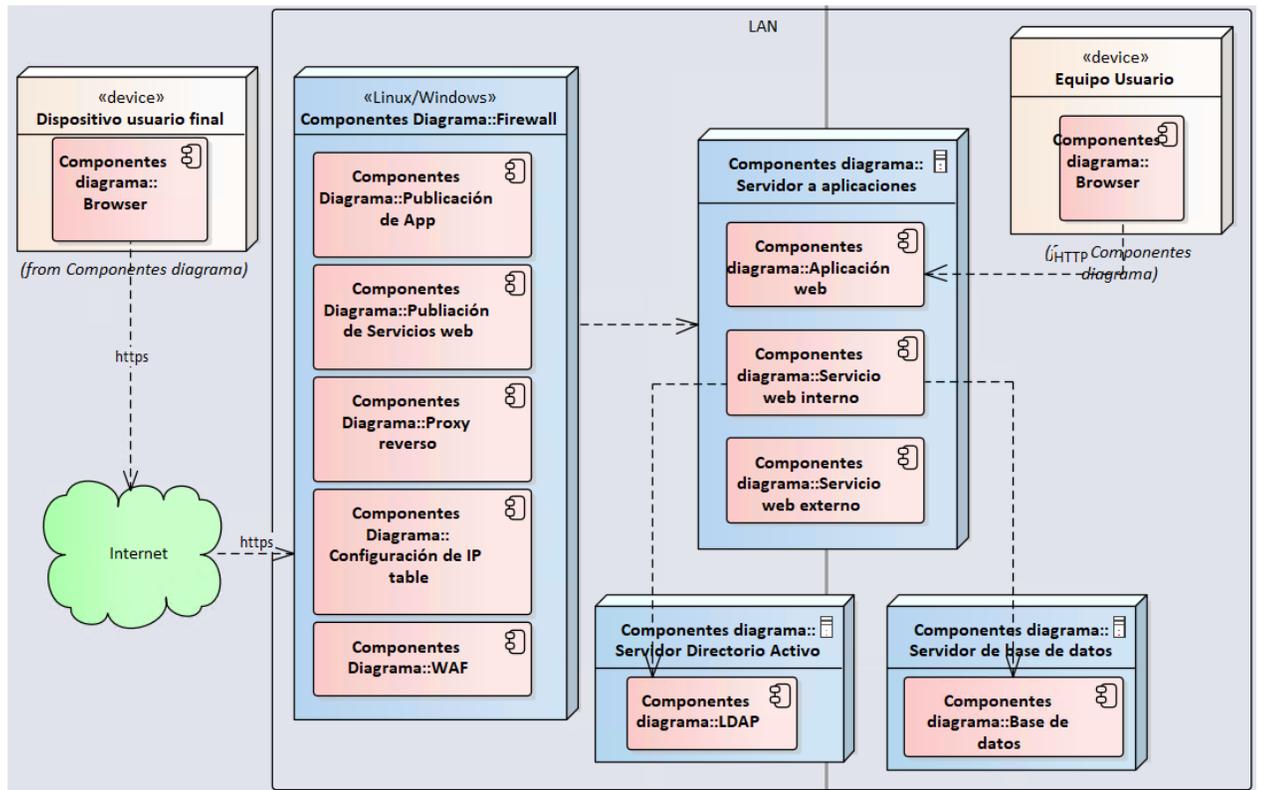


Ilustración 106. Arquitectura de aplicaciones web, diagrama de despliegue

6.2.11.2. Arquitectura de referencia para aplicaciones móviles

Diagrama de componentes

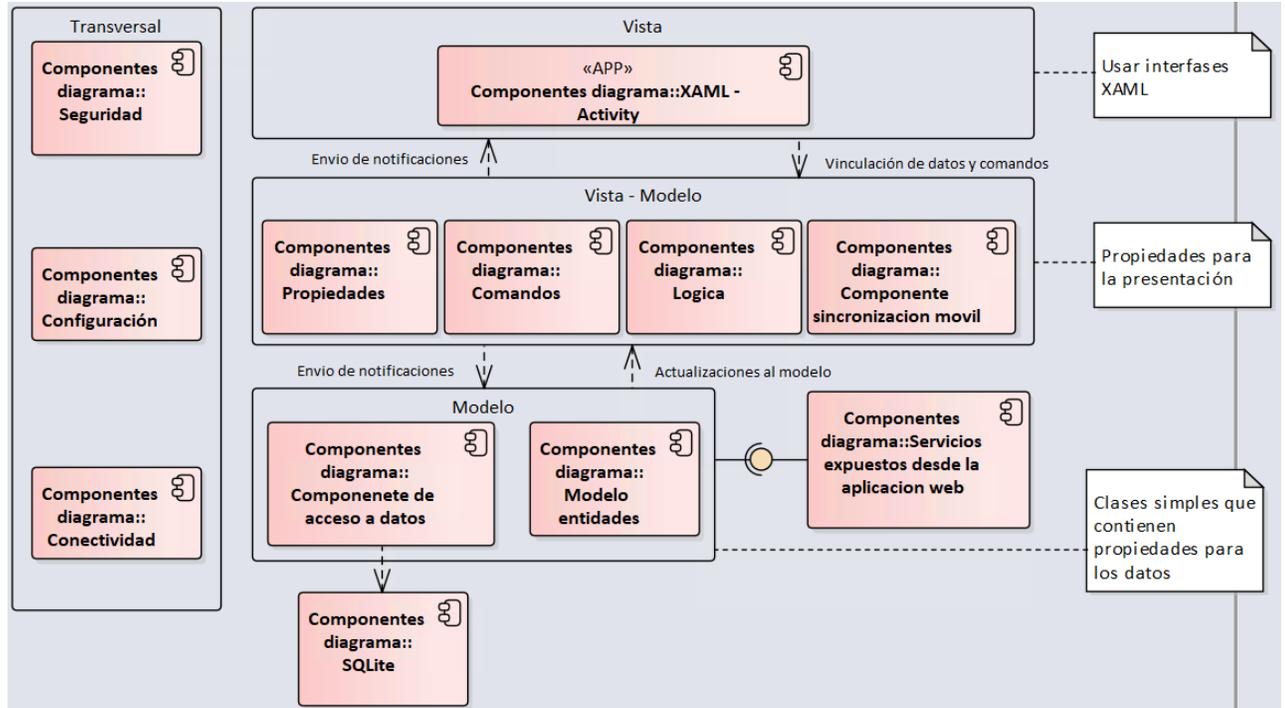


Ilustración 107. Arquitectura móvil, diagrama de capas

Diagrama de distribución

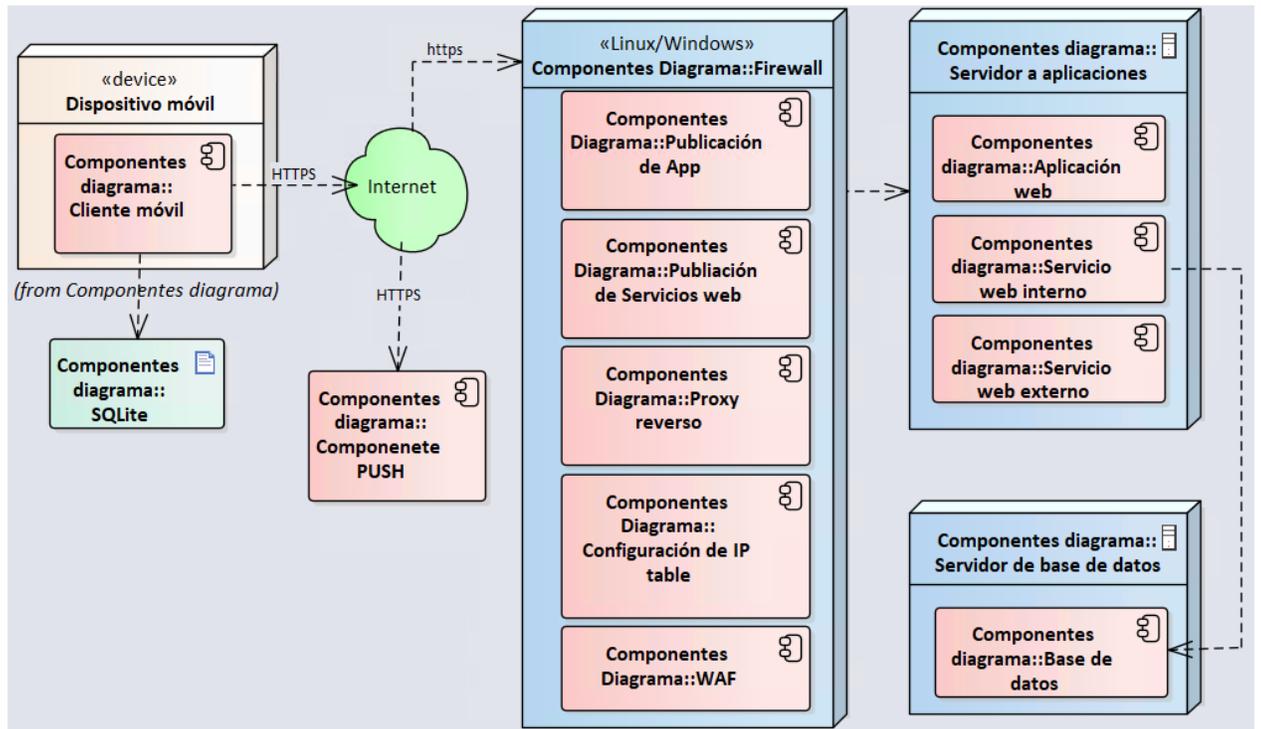


Ilustración 108. Arquitectura de aplicaciones móviles. Diagrama de distribución

6.2.11.3. Arquitectura referencia para de integración

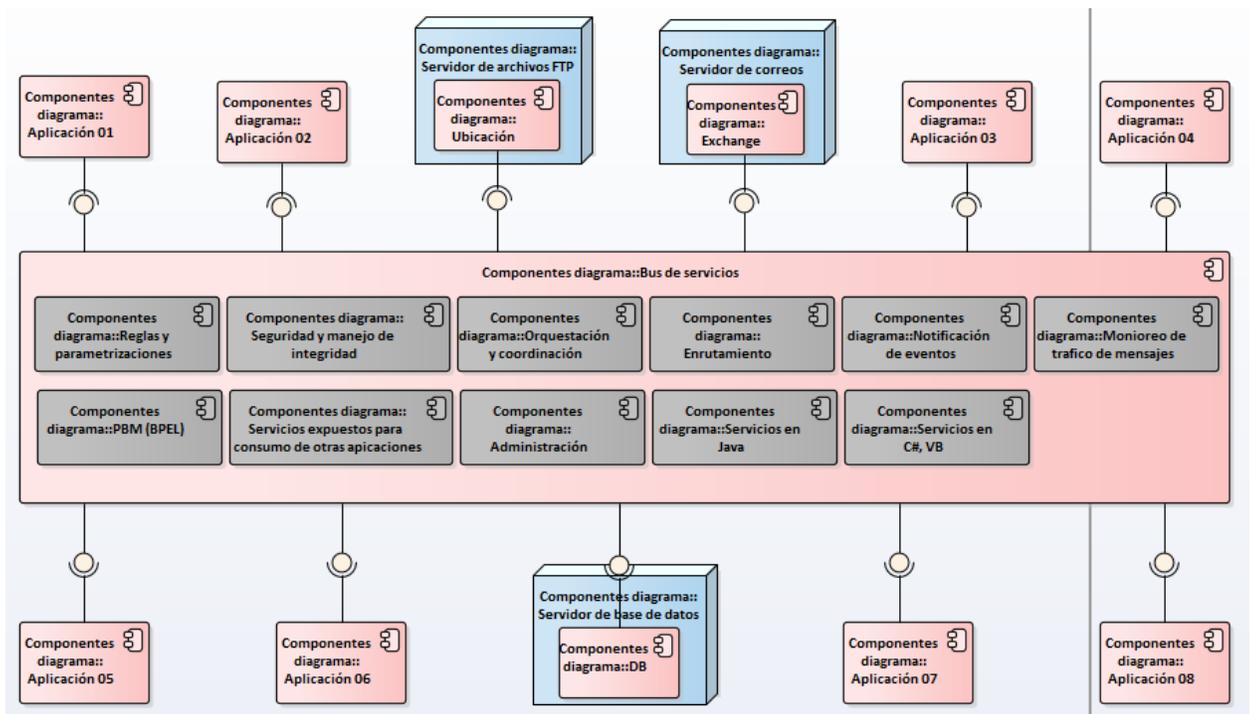


Ilustración 109. Arquitectura de integración

Observaciones

- Todos los servicios deben quedar registrados en el catálogo de servicios de la capa de aplicaciones y datos
- La implementación de un servicio debe obedecer a al diseño entregado por arquitectura, ya que es en este punto donde se evalúan los atributos de calidad
- Los servicios deben ser implementados siempre y cuando la capacidad tecnológica lo permita
- Algunos servicios serán de implementación interna o externa según la necesidad del cliente y del negocio
- Antes de desarrollar un servicio se debe validar en el catálogo de servicios si este ya fue creado previamente
- Antes de modificar un servicio se debe mapear todas las aplicaciones y otros servicios que hacen uso de este, con el fin de minimizar e involucrar a todos los interesados

Ilustración 110. Observaciones sobre la arquitectura de integración

6.2.11.4. Arquitectura de referencia para infraestructura

Para visualizar la arquitectura de referencia de infraestructura se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas de referencia. Si tiene la aplicación de Enterprise Architect abrir el archivo Arquitectura de Referencia.EAP, en caso contrario ingresar a la carpeta Arquitectura de referencia y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

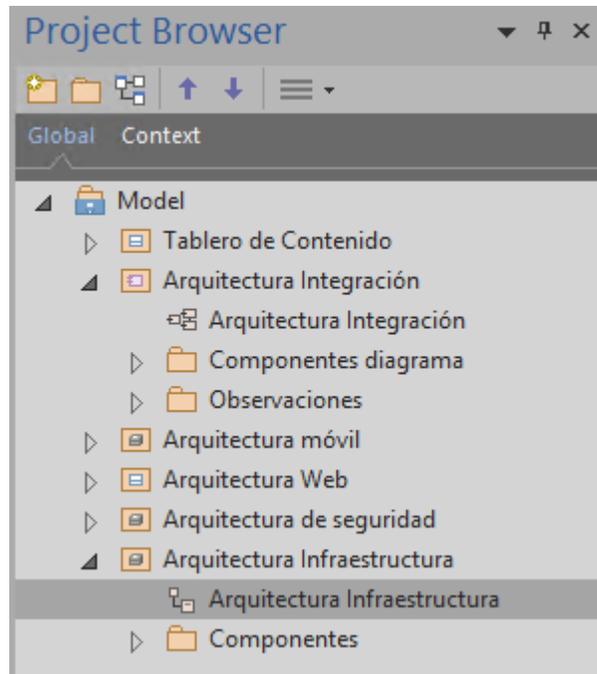


Ilustración 111. Navegación a la Arquitectura de referencia de infraestructura

6.2.11.5. Arquitectura de referencia para seguridad

Para visualizar la arquitectura de referencia de seguridad se debe navegar en la siguiente ruta: Repositorio de Arquitectura -> Arquitecturas de referencia. Si tiene la aplicación de Enterprise Architect abrir el archivo Arquitectura de Referencia.EAP, en caso contrario ingresar a la carpeta Arquitectura de referencia y abrir el archivo index.html

Al interior del archivo se debe navegar con la siguiente ruta:

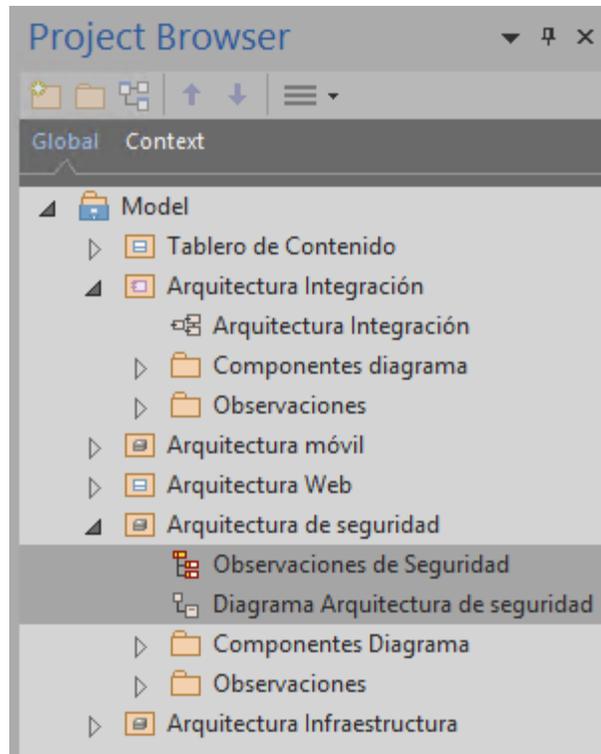


Ilustración 112. Navegación a la Arquitectura de referencia de seguridad

6.3. Objetivo 3- Implementación del marco de trabajo adaptado

El marco de trabajo adaptado se evaluó en un proyecto para una empresa a la cual no se le referenciará el nombre por temas de confidencialidad. En el proceso de la prueba piloto se siguió cada uno de los pasos especificados en el punto [7.2. Marco de trabajo adaptado](#), con el fin de validar la complejidad y eficiencia de las actividades y documentos planteados en cada fase.

La implementación del marco de trabajo adaptado se simuló en una iniciativa para una empresa de energía de Colombia. Se tomó este proyecto porque en una primera fase se idéntico que los procesos no eran oficiales y las personas trabajaban e implementaban macros y bases de datos como consideraban prudente, sin aprobación de un área encargada o los líderes del proceso. Como resultado a lo anterior la primera fase no cumplió las expectativas del cliente y quedaron requisitos por fuera del alcance.

Nota aclaratoria: Las plantillas para diagramar la arquitectura fueron construidas usando la aplicación de *Enterprise Architect*, sin embargo, la

implementación de la metodología se puede realizar sobre cualquier herramienta que permite la diagramación y trazabilidad de los componentes. La aplicación cuenta con un trial por 30 días que puede ser descargado de la página del fabricante⁸² o en caso de no querer descargar la aplicación se tienen versiones en HTML, las cuales pueden ser visualizadas usando un explorador.

6.3.1. Documentos generados

Los documentos generados en el piloto se encuentran ubicados en el repositorio de arquitectura en la carpeta Arquitecturas -> Arquitectura campañas.

Este sitio almacena toda la información de arquitectura, adicionalmente es consecuente con el marco de trabajo adaptado planteado. Se adicionarán unas imágenes como evidencia de su desarrollo, sin embargo, el detalle debe ser consultado en el repositorio de arquitectura.

6.4. Objetivo 4 - Conclusiones del uso del marco de trabajo adaptado

Las conclusiones se detallarán en la siguiente sección del documento

⁸² <http://www.sparxsystems.com/products/ea/trial/request.html>

7. CONCLUSIONES

Luego de la implementación del marco de trabajo adaptado en un proyecto real se identifican diferentes fortalezas y oportunidades de mejora sobre la metodología, las cuales serán detalladas a continuación.

7.1.1. Cómo apoya la arquitectura el marco de trabajo adaptado

- Luego de oficializar y conocer el proceso de la gestión de campañas fue más fácil identificar el alcance de la solución y las reglas de negocio que se debían implementar, ya que los analistas tuvieron el tiempo suficiente para oficializar los procedimientos y resaltar como el desarrollo de las funcionalidades en la herramienta les facilitaba su labor desde las oficinas para guiar actividades en campo.
- Se pudo identificar la alineación de la estrategia de la empresa de energía con el desarrollo de la arquitectura, ya que se estaba apalancando el objetivo de evitar pérdidas.
- Al mapear el proceso con los sistemas de información que se estaban usando se identificó que este estaba siendo soportado por un conjunto de macros en Excel y posteriormente se debía manipular datos, lo que infringía los principios de arquitectura y aumentaba los riesgos de los informes entregados a otros procesos.
- Se identificó que se reduce la complejidad del ciclo de vida de las soluciones de TI, ya que los talleres ayudan a entender los procesos de forma más dinámica y a conocer las expectativas de los interesados. Adicionalmente el desarrollo ágil de productos permite que la solución sea accedida por los usuarios funcionales.
- Con los talleres se pudo identificar que la interacción cercana con los funcionales brinda mayor confiabilidad y seguridad por parte de los participantes, lo que les facilita la entrega de información y de esta manera se cumple el objetivo de conocer el flujo real de los datos de los procesos y los componentes que intervienen.

- Se reduce los riesgos de negocio y de TI, ya que con los talleres se puede conocer la visión global de los directivos, y la visión operativa de los analistas funcionales. Ambas expectativas se complementan y de esta forma se entrega una solución integral.
- Se evidenció cómo se optimiza los procesos de negocio a través de la integración de soluciones, ya que los directivos y analistas funcionales resaltaron que, con la automatización de los mismos, las intervenciones humanas disminuían y la mayor parte del tiempo que se dedicaba a tareas operativas ya se iba a invertir en análisis de información.
- Se evidencia en los procesos cómo el desarrollo de soluciones tecnológicas ayuda a alcanzar la estrategia de la compañía, ya que cada componente de la arquitectura está relacionado y mapeado. Los directivos y analistas funcionales conocieron estas relaciones a partir de las vistas generadas, donde se evidencia la situación actual y la situación deseada.

7.1.2. Fortalezas del marco de trabajo adaptado

- Tener una herramienta para diagramar permite garantizar la trazabilidad entre los componentes de la arquitectura, permitiendo identificar cómo se afecta alguno de ellos con modificaciones o adiciones en las arquitecturas futuras.
- Los talleres permiten a los interesados entrar en más confianza con el equipo de trabajo y expresar sus ideas y expectativas de forma natural.
- El uso de historias de usuarios permite conocer a un nivel más alto las necesidades de los usuarios y el complemento con los requisitos ayuda a identificar las reglas de negocio claves que deben tenerse en cuenta en la materialización de la solución.
- Los lenguajes de modelado se complementan entre sí, ya que Archimate 3.0 permite una visión completa de cada una de las capas, y a medida que se navega entre ellas se puede ver el detalle de los procesos usando BPMN 2.0 y de los sistemas de información usando UML 2.0.

- El marco de trabajo adaptado respetó los lineamientos que entrega Togaf 9.1 para integrarse con otras metodologías o marcos de trabajo como la ITIL y las directrices del *PMBOOK*. Ya que el desarrollo de proyectos obedeció a los lineamientos de la PMO de la empresa de energía, al igual que la entrega del servicio a la mesa de ayuda.
- La trazabilidad de en los componentes de arquitectura permite identificar la mayoría de los interesados de forma temprano, lo que permite gestionar sus expectativas y evaluar si sus necesidades si están en el alcance de la solución.

7.1.3. Oportunidades de mejora

Básicamente las oportunidades de mejora permiten que el marco de trabajo adaptado sea más ágil, es decir, al implementar la metodología se identificó que algunas actividades pueden ser fusionadas y diligenciadas en un solo artefacto, de esta forma se evita que información sea duplicada y confunda a los interesados que acceden a los entregables generados. Cabe resaltar que para efectos del piloto se siguieron todas las directrices como el marco de trabajo adaptado las sugería.

- Los objetivos estratégicos y requerimiento de alto nivel se deben diligenciar en el documento de “Definición de arquitectra.EAP”, por lo tanto, pueden ser omitidos en el documento de la visión de la solución.
- El desarrollo de todos los diagramas no debe ser obligatorio, es decir, solo se desarrolla si genera valor para la implementación de la solución o determinar el alcance. Por ejemplo, si en una solución las transacciones solo tienen tres estados no es necesario desarrollar un diagrama de estados.
- En el documento plan de implementación los hitos deben ser responsabilidad del gerente de proyectos, de esta forma los limites queda correctamente definidos y no se entra en conflicto con la sección de entregables.

- Las historias de usuarios ya existen en el documento “Definición de arquitectra.EAP”, por lo que se deben suprimir del documento plan de implementación
- Los entregables se encuentran en el documento de declaración del trabajo de arquitectura, visión de la solución e implementación y migración, todas con tablas o títulos diferentes, se sugiere unificar el contenido y administrarse solo desde el documento de visión, de esta forma se garantiza que solo es creada y modificada desde una sola fuente de datos.
- Las capacidades se encuentran en el documento de visión de la solución e implementación y migración, ambas con tablas diferentes, se sugiere unificar la tabla y administrarse solo desde el documento de visión, de esta forma se
- garantiza que solo es creada y modificada desde una sola fuente de datos. Las brechas se encuentran en el documento “Definición de arquitectra.EAP” y el plan de implementación y migración, se sugiere que la creación y modificación de estas se desarrollen en el documento de “Definición de arquitectra.EAP”, con el fin de garantizar la creación y modificación de componentes desde una única fuente de datos.

8. REFERENCIAS

- Aranda software. (2010). http://arandasoft.com/webcast-antteriores/catalogo_%20servicioV3-2.pdf. Aranda software. Obtenido de http://arandasoft.com/webcast-antteriores/catalogo_%20servicioV3-2.pdf
- Azevedo, C., Almeida, J., Sinderen, M., Quartel, D., & Guizzardi, G. (2011). An Ontology-Based Semantics for the Motivation Extension to ArchiMate. *IEEE Explorar Digital Export*. Obtenido de <http://ieeexplore.ieee.org/abstract/document/6037557/>
- Band, I., Jonkers, H., Proper, E., Quartel, D., & Lankhorst, M. (2017). Using the TOGAF® 9.1 Framework with the ArchiMate® 3.0 Modeling Language.
- Barker, R. (1990). *El modelo entidad-relación CASE METHOD*. Massachusetts E.U.A: ADDISON-WESLEY / DIAZ DE SANTOS.
- Beck, K. (1999). *Extreme Programming Explained. Embrace Change*. Boston: Addinson-Wesley.
- Benavides, D., & Galindo, J. (2013). Variability management in an unaware software product line company: an experience report. *Proceeding. VaMoS '14 Proceedings of the Eighth International Workshop on Variability Modelling of Software-Intensive Systems*.
- Cabrera, D., & Lopez, G. (2014). *Diseño de una arquitectura empresarial basada en el framewor Togaf para INSOTEC*.
- Carlos L.B. Azevedo, M.-E. I. (2015). Modeling resources and capabilities in enterprise architecture: A well-founded ontology-based proposal for ArchiMate. *Information Systems*, 235-262.
- Chiprianov, V., Kermarrec, Y., & Rouvrais, S. (2012). Extending Enterprise Architecture Modeling Languages: application to telecommunications service creationg. *SAC '12 Proceedings of the 27th Annual ACM Symposium on Applied Computing*, 1661-1666.
- Dante Carrizo, O. D. (2014). Systematizing requirements elicitation technique selection. *Information and Software Technology*.
- Dave Brown, P. C. (12 de 11 de 2015). *Arquitectura empresarial para Ingenieros de Sistemas*. Obtenido de The Rational Edge: <http://www.ibm.com/developerworks/ssa/rational/library/edge/09/jun09/enterpri searchitecture/>
- Duarte, G. (2016). *Arquitectura Propuesta para un Servicio Web Completo: Metodolog´ia de Desarrollo e Implementaci3n*. Bogota: Universidad Distrital Francisco Jos´e de Caldas. Obtenido de <http://repository.udistrital.edu.co/bitstream/11349/2809/1/DuarteVegaGabrielEdu ardo2016.pdf>
- Duran, A., & Bernardez, B. (2002). *Metodolog´ia para la Elicitaci3n de Requisitos de Sistemas Software*. Sevilla: Universidad de Sevilla. Obtenido de https://uvirtual.unet.edu.ve/pluginfile.php/177120/mod_resource/content/0/metodologia_ elicitation23.pdf
- Ernst, A. M. (2008). Enterprise architecture management patterns. *PLoP '08 Proceedings of the 15th Conference on Pattern Languages of Programs*.
- Escalona, M. J., & Nora, K. (2002). *Ingenier´ia de Requisitos en Aplicaciones para la Web – Un estudio comparativo*. Sevilla: Departamento de Lenguajes y Sistemas Inform´aticos, Escuela T´ecnica Superior de Ingenier´ia Inform´atica, Universidad de Sevilla. Obtenido de <https://www.lsi.us.es/docs/informes/LSI-2002-4.pdf>

- Freund, J., Rücker, B., & Hitpass, B. (2014). *BPMN 2.0 Manual de Referencia y Guía Práctica* (Cuarta edición ed.). Santiago de Chile, Chile: Dimacofi.
- Gasca, J., & Zaragoza, R. (2014). *Designpedia - 80 herramientas para construir tus ideas*. Madrid: LID Editorial.
- Godoy, D., Belloni, E., Kotynsky, H., Dos Santos, H., & Sosa, E. (2014). Simulando proyectos de desarrollo de software administrados con Scrum. *Workshop de Investigadores en Ciencias de la Computación*, 485.
- Grupo ISSI. (12 de Noviembre de 2003). *Metodologías Ágiles en el Desarrollo de Software*. Obtenido de Metodologías Ágiles en el Desarrollo de Software: http://s3.amazonaws.com/academia.edu.documents/33257239/METODOLOGIAS_AGILES.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1489849732&Signature=7p1E5WcAhmCAyMae4Ykbg9fZpE%3D&response-content-disposition=inline%3B%20filename%3DMetodologias_Agiles_en_el_De
- Harmon, P. (2007). *Business Process Change (Second Edition) A Guide for Business Managers and BPM and Six Sigma Professionals (chapter 9 – Modeling Processes)*. Morgan Kaufmann. Obtenido de <http://www.sciencedirect.com/science/article/pii/B9780123741523500409>
- Hinkelmann, K., Gerber, A., Dimitris, K., Thoenssen, B., Merwe, A. v., & Woitsch, R. (2015). A new paradigm for the continuous alignment of business and IT: Combining enterprise architecture modelling and enterprise ontology. *Computers in Industry*.
- Hitpass, B. (2017). *Business Process Management (BPM) Fundamentos y conceptos de Implementación*. Santiago de Chile: BPM Center.
- Izaurrealde, M. P. (2013). *Caracterización de Especificación de Requerimientos en entornos Ágiles: Historias de Usuario*. Córdoba: Universidad Tecnológica Nacional. Obtenido de http://www.institucional.frc.utn.edu.ar/sistemas/lidicalso/pub/file/Tesis/Anteproyecto_Requerimientos_en_Metodolog%C3%ADas_Agiles.pdf
- Jonkers, H., Berg, H., Iacob, M., & Quartel. (2010). ArchiMate® Extension for Modeling the TOGAF™ Implementation and Migration Phases. *Open Group*. Obtenido de http://pinkelephantcom.vanharen.net/Player/eKnowledge/w111_-_archimate_extension_for_modeling_the_togaf_implementation_and_migration_phases.pdf
- Josey, A. (2013). *TOGAF © V9.1 - Guía de Bolsillo* (Primera ed.). (Z. Van Haren Publishing, Ed., & L. Infanti, Trad.) Wilco, Amersfoort, Países Bajos: Van Haren Publishing.
- Kimmel, P. (2008). *Manual de UML Guía de aprendizaje*. Mexico: Mc Graw Hill. Obtenido de https://docs.google.com/file/d/0B_DCrU4iv2eiSFYwUGxTVmQzYnM/edit?pli=1
- Kurpjuweit, S., & Winter, R. (2009). Concern-oriented business architecture engineering. *In Proceedings of the 2009 ACM symposium on Applied Computing (SAC '09)*, 265-272.
- Larman, C. (2003). *UML y Patrones*. Madrid: PrenticeHall.
- Mario R. Barbacci, R. J. (2003). Quality Attribute Workshops (QAWs), Third Edition. *Software Engineering Institute*, 38. Obtenido de <http://www.sei.cmu.edu/reports/03tr016.pdf>
- Mesa, A., Lochmuller, C., & Tabares, M. (2014). Comparativo entre herramientas BPMN. *Revista Soluciones de Postgrado EIA*, 6(12), 95-108.
- Michele Chinosi, A. T. (2012). BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 124-134.

- Microsoft. (2016). *MSDN - Diagramas de capas: Referencia*. Obtenido de MSDN - Diagramas de capas: Referencia: <https://msdn.microsoft.com/es-co/library/dd409462.aspx>
- Ministerio de ciencia, tecnología y telecomunicaciones (Costa Rica). (s.f.). *Design Thinking Pensamiento del diseño*. Costa Rica: micitt. Obtenido de https://www.innovacion.cr/sites/default/files/article/adjuntos/herramientas_practicas_para_innovacion_1.0_design_thinking_1.pdf
- Napoli, J., & Kaloyanova, K. (2011). An integrated approach for RUP, EA, SOA and BPM implementation. *CompSysTech '11 Proceedings of the 12th International Conference on Computer Systems and Technologies*, 63-68.
- Nasrin Dastranj Mamaghani, F. M. (2012). Customer oriented enterprise IT architecture framework. *Telematics and Informatics*, 219-232.
- Nuseibeh, B., & Easterbrook, S. (2000). Requirements Engineering: A Roadmap. *Proceeding ICSE '00 Proceedings of the Conference on The Future of Software Engineering*, 35-46.
- Object Management Group (OMG). (2011). *Business Process Model and Notation (BPMN)*. OMG. Obtenido de Business Process Model and Notation (BPMN).
- Object Management Group (OMG). (2016). *Object Management Group Business Process Model and Notation*. Obtenido de Object Management Group Business Process Model and Notation: <http://www.bpmn.org/>
- OMG. (31 de Mayo de 2016). *Business Process Model & Notation™ (BPMN™)*. Obtenido de Business Process Model & Notation™ (BPMN™): <http://www.omg.org/bpmn/index.htm>
- Open Group. (2016). *Welcome to the ArchiMate® 3.0 Specification, an Open Group Standard*. Obtenido de Welcome to the ArchiMate® 3.0 Specification, an Open Group Standard: <http://pubs.opengroup.org/architecture/archimate3-doc/toc.html>
- Open Group. (2016). *What is New in ArchiMate® 3.0?* Obtenido de What is New in ArchiMate® 3.0?: <http://www.opengroup.org/subjectareas/enterprise/archimate/3.0-whats-new>
- Pais, J. R. (2013). *Business Process Management*. BPM Teca.
- Pan, D., Zhu, D., & Johnson, K. (2001). *Requirements Engineering Techniques*. Canada: Department of Computer Science. University of Calgary.
- Pereira, J., Serpa, N., & Rivas, M. (2004). *FACTORES DE EXITO EN PROYECTOS DE DESARROLLO DE SOFTWARE*. Obtenido de [www.researchgate.net: https://www.researchgate.net/profile/Narciso_Cerpa/publication/233754771_Factores_de_xito_en_proyectos_de_desarrollo_de_software_analisis_de_la_industria_cilena_del_software/links/0912f512a2b3c6af6e000000.pdf](https://www.researchgate.net/profile/Narciso_Cerpa/publication/233754771_Factores_de_xito_en_proyectos_de_desarrollo_de_software_analisis_de_la_industria_cilena_del_software/links/0912f512a2b3c6af6e000000.pdf)
- Pilone, D., & Pitman, N. (2005). *UML 2.0 In a Nutshell* (Vol. Primera Edición). Gravenstein Highway North, United States of America: O'Reilly.
- Quintana, A. y. (2006). Metodología de Investigación Científica Cualitativa. *Psicología: Tópicos de actualidad*, 47-84.
- Qumer Gill, A. (2015). Agile enterprise architecture modelling: Evaluating the applicability and integration of six modelling standards. *Journal of Systems and Software*, 196-206. Obtenido de <http://www.sciencedirect.com/consultaremot/upb.edu.co/science/article/pii/S0950584915001251>
- Rumbaugh, J., Jacobson, I., & Grady, B. (2004). *The Unified Modeling Language Reference manual Second Edition*. Boston: Addison-Wesley. Obtenido de

- http://www.utd.edu/~chung/Fujitsu/UML_2.0/Rumbaugh--UML_2.0_Reference_CD.pdf
- Ruparelia, N. (2010). Software Development Lifecycle. *ACM SIGSOFT Software Engineering*, 8-13. Obtenido de <http://dl.acm.org.consultaremota.upb.edu.co/citation.cfm?id=1764814&CFID=905098531&CFTOKEN=32444828>
- Salgado, C., Machado, R., & Suzana, R. (2013). Modeling the alignment between business and IS/IT: a requirements engineering perspective. *Proceeding. SAC '13 Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 1309-1311.
- Santos, A., Kroll, J., Sales, A., Fernandes, P., & Wild, D. (2016). Investigating the Adoption of Agile Practices in Mobile Application Development. *ICEIS*, 490.
- Šaša, A., & Krisper, M. (2011). Enterprise architecture patterns for business process support analysis. *Journal of Systems and Software*, 1480–1506. Obtenido de <http://www.sciencedirect.com.consultaremota.upb.edu.co/science/article/pii/S0164121211000689>
- Satpathy, T. (2016). *Cuerpo de conocimiento de scrum*. Arizona: SCRUMstudy.
- Schwaber, K., & Sutherland, J. (Julio de 2013). *Scrum Guides*. Obtenido de La Guía de Scrum: <http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-es.pdf>
- Scrum.org. (2005). *Metodologías Scrum*. Obtenido de Metodologías Scrum: <http://metodologiascrum.readthedocs.io/en/latest/Scrum.html>
- Serrano, M., & Pilar, B. (2014). *Design Thinkin lidra el presente, crea el futuro*. Madrid: ESIC Editorial.
- Sessions, R. (2007). *A Comparison of the Top Four Enterprise-Architecture Methodologies*. Obtenido de msdn.microsoft.com: <https://msdn.microsoft.com/en-us/library/bb466232.aspx>
- Shahzeydi, M., & Gandomani, T. (2016). Adding lean principles to agile software development: A case study report. *International Journal of Software Engineering and Technology*, 12.
- Shi, Z., Zenga, X., Zhang, T., HuangSong, Qi, Z., Li, H., . . . Zhong, S. (2016). Bidirectional transformation between BPMN and BPEL with. *Computers and Electrical Engineering*.
- Singh P.M., S. M. (2017). Reference architecture for integration platforms. *IEEE E*, 113-122.
- Software Engineering Institute - SEI. (10 de 11 de 2015). *Quality Attribute Workshop*. Recuperado el 10 de 11 de 2015, de Quality Attribute Workshop: <http://www.sei.cmu.edu/architecture/tools/establish/qaw.cfm>
- Spark System. (06 de Noviembre de 2015). *Tutorial UML 2*. Recuperado el 06 de Noviembre de 2015, de Tutorial UML 2: http://www.sparxsystems.com.ar/resources/uml2_tutorial.html
- The Open Group. (2011). *TOGAF® Version 9.1*. E.U: Open Group Standard.
- The Open Group. (2013). *ArchiMate 2.1 Specification*. Amersfoort , Paises Bajos, Holanda: Van Haren.
- Tupper, C. D. (2011). *Data Architecture*. (M. Kaufmann, Ed.) Burlington, USA: Morgan Kaufmann .
- Weilkiens, T. (2008). *Systems Engineering with SysML/UML*. Elsevier Inc. Obtenido de http://ac.els-cdn.com/B9780123742742000031/3-s2.0-B9780123742742000031-main.pdf?_tid=00dd6278-03a1-11e7-a40a-00000aacb362&acdnat=1488937749_44fd30dda8b594a60bb0b855069a19fa

- Weilkiens, T., & Oestereich, B. (2007). *UML 2 Certification Guide Fundamental and Intermediate Exams*. San Francisco, California, USA: Morgan Kaufmann.
- Weilkiens, T., Weiss, C., & Grass, A. (2011). *Business Process Management – Fundamental Level*. Waltham, USA: Morgan Kaufmann.
- Weilkiens, T., Weiss, C., & Grass, A. (2011). *OCEB Certification Guide (CHAPTER 6 Modeling Business Processes Using BPMN)*. Elsevier Inc. Obtenido de <http://www.sciencedirect.com.consultaremota.upb.edu.co/science/article/pii/B9780123869852000022>
- Wellsandt, S., Hribernik, K., & Dieter T, K. (2014). Qualitative Comparison of Requirements Elicitation Techniques that are Used to Collect Feedback Information about Product Use. *Procedia CIRP*, 212-217.
- Würfel, D., Lutz, R., & Diehl, S. (2015). Grounded Requirements Engineering: An Approach to Use Case Driven Requirements Engineering. *Journal of Systems and Software*.
- Yagüe, A., & Garbajosa, J. (2009). *Comparativa práctica de las pruebas en entornos tradicionales y ágiles*. Madrid: Revista Española de Innovación, Calidad e Ingeniería del Software. Obtenido de <https://pdfs.semanticscholar.org/606f/8965493e2a470a98ce285ec7c85535e0a879.pdf>
- Zapata, C., & Arango, F. (2004). ALINEACIÓN ENTRE METAS ORGANIZACIONALES Y ELICITACIÓN DE REQUISITOS DEL SOFTWARE. *Revista de la facultad de minas - Universidad Nacional de Colombia - Sede Medellin*, 101-110. Obtenido de https://www.researchgate.net/profile/Carlos_Zapata7/publication/26460714_Alineacion_entre_metas_organizacionales_y_elicitacion_de_requisitos_del_software/links/0046351a3642f5fb6c000000.pdf

ANEXO 1

Tabla de Entregables por Fase

Fase	Documento	Plantilla	Almacenamiento
Preliminar	Declaración del trabajo de arquitectura	Repositorio de arquitectura -> Plantillas -> Declaración del trabajo de arquitectura.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Documentos de gestión -> Declaración del trabajo de arquitectura.docx
	Principios de arquitectura	Repositorio de arquitectura -> Documentación de apoyo -> Principios de arquitectura.docx	Repositorio de arquitectura -> Principios -> Principios de arquitectura.docx
Fase A - Visión de la arquitectura	Declaración del trabajo de arquitectura	Repositorio de arquitectura -> Plantillas -> Declaración del trabajo de arquitectura.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Documentos de gestión -> Declaración del trabajo de arquitectura.docx
	Visión de la arquitectura	Repositorio de arquitectura -> Plantillas -> Visión de arquitectura.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Documentos de gestión -> Visión de arquitectura.docx.
	Principios de arquitectura	Repositorio de arquitectura -> Documentación de apoyo -> Principios de arquitectura.docx	Repositorio de arquitectura -> Principios -> Principios de arquitectura.docx

	Definición de arquitectura	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase B - Arquitectura de negocio	Arquitectura del Negocio	Repositorio de arquitectura -> Plantillas -> Definición de arquitectur.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectur.EAP -> Arquitectura Base Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Arquitectura Destino
	Historias de usuario	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Historias de Usuario
	<i>Product Backlog</i>	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Historias de Usuario

	Requerimientos	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Requisitos
	Catálogo de servicios	Repositorio de arquitectura -> Plantillas -> Catalogo de arquitectura.EAP	Repositorio de arquitectura -> Catalogos -> Catalogo de arquitectura.EAP
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase C - Arquitectura de sistemas de información	Arquitectura de sistemas de información - Línea base	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Arquitectura Base
	Arquitectura de sistemas de información - arquitectura destino	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Arquitectura Destino
	Brechas	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Análisis de brechas

	Historias de usuario	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Historias de Usuario
	Prototipos		Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Prototipos
	Requisitos funcionales	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Requisitos -> Requisitos funcionales
	Requisitos no funcionales	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Requisitos -> Requisitos no funcionales
	Modelo de dominio	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Modelo de dominio
	Modelo entidad relación		Visible desde el motor de base de datos

	Diagrama de componentes	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Diagrama de componentes
	Diagrama de capas	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Diagrama de capas
	Diagrama de despliegue	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Diagrama de despliegue
	Diagrama de estados	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Diagrama de estados
	Catálogo de servicios	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Catalogos -> Catalogo de arquitectura.EAP
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx

Fase D - Arquitectura tecnológica	Arquitectura tecnológica - Línea base	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Arquitectura Base
	Arquitectura tecnológica - arquitectura destino	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Arquitectura Destino
	Brechas	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Análisis de brechas
	Catálogo de servicios	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Catálogos -> Catalogo de arquitectura.EAP
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase E - Oportunidades y soluciones	Mapa de ruta	Repositorio de arquitectura -> Plantillas -> RoadMap.xlsx	Repositorio de arquitectura -> Mapas de ruta -> RoadMap.xlsx
	Limitaciones	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Motivación

	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase F - Planificación de la migración	Plan de implementación y migración	Repositorio de arquitectura -> Plantillas -> Plan de implementación y migración.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Documentos de gestión -> Plan de implementación y migración.docx
	Mapa de ruta	Repositorio de arquitectura -> Plantillas -> RoadMap.xlsx	Repositorio de arquitectura -> Mapas de ruta -> RoadMap.xlsx
	Definición de arquitectura	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase G - Gobierno de la implementación	Solicitud de cambio	Repositorio de arquitectura -> Plantillas -> Control de cambios.docx	Repositorio de arquitectura -> Arquitecturas -> Arquitectura a trabajar -> Controles de cambio -> Control de cambio_AAAAMMDD.docx
	Paquete de trabajo o incremento		Implementación en servidores
	<i>Sprint backlog</i>	Repositorio de arquitectura -> Plantillas -> Sprint backlog_XX.xlsx	Repositorio de arquitectura -> Arquitecturas -> Arquitectura a trabajar ->

			Sprint Backlog -> Sprint backlog_XX.xlsx
	Evaluación de conformidad	Repositorio de arquitectura -> Plantillas -> Evaluación de conformidad.xlsx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Auditorias -> Evaluación de conformidad.docx
	Acta	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Fase H - Gestión del cambio de la arquitectura	Definición de arquitectura	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP
	Acta - Respuesta a la solicitud del cambio	Repositorio de arquitectura -> Plantillas -> Acta de Reunión.docx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Actas -> Acta de reunión.docx
Gestión de requerimientos	Requisitos funcionales	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Requisitos -> Requisitos funcionales
	Requisitos no funcionales	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP ->

			Requisitos -> Requisitos no funcionales
	Modelo de dominio	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP -> Diseño -> Modelo de dominio
	Evaluación de conformidad	Repositorio de arquitectura -> Plantillas -> Evaluación de conformidad.xlsx	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Auditorias -> Evaluación de conformidad.docx
	Definición de arquitectura	Repositorio de arquitectura -> Plantillas -> Definición de arquitectura.EAP	Repositorio de arquitectura -> Arquitectura -> Arquitectura a trabajar -> Definición de arquitectura.EAP