

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
ESCUELA DE POSGRADO



**Business Consulting: Aplicaciones de marcos de referencia para el desarrollo
de software en MS4M S.A.C.**

**TESIS PARA OBTENER EL GRADO ACADÉMICO DE MAESTRA EN
GERENCIA DE TECNOLOGÍAS DE INFORMACIÓN OTORGADO POR
LA PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**

QUE PRESENTA:

Carmela Romero Aguilar

**TESIS PARA OBTENER EL GRADO ACADÉMICO DE MAESTRO EN
GERENCIA DE TECNOLOGÍAS DE INFORMACIÓN OTORGADO POR
LA PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**

QUE PRESENTA:

Andric Alejandro Vargas Alarcón

Richar Sandro Balboa Zegarra

Victor Alvarado Sanchez

Victor Hugo Plasencia Sifuentes

ASESOR

Luis Alfredo Negrón Naldos

Surco, agosto 2024

Declaración Jurada de Autenticidad

Yo, Luis Alfredo Negrón Naldos, docente del Departamento Académico de Posgrado en Negocios de la Pontificia Universidad Católica del Perú, asesor(a) de la tesis/el trabajo de investigación titulado: " Aplicaciones de marcos de referencia para el desarrollo de software en MS4M S.A.C".

del/de la autor(a)/ de los(as) autores(as)

- Alvarado Sanchez Victor, DNI: 06807168
- Balboa Zegarra Richar Sandro, DNI: 09931967
- Plasencia Sifuentes Victor Hugo, DNI: 10588536
- Romero Aguilar Carmela, DNI: 46137783
- Vargas Alarcón Andric Alejandro, DNI: 46676673

dejo constancia de lo siguiente:

- El mencionado documento tiene un índice de puntuación de similitud de 12%. Así lo consigna el reporte de similitud emitido por el software *Turnitin* el 23/09/2024.
- He revisado con detalle dicho reporte y confirmo que cada una de las coincidencias detectadas no constituyen plagio alguno.
- Las citas a otros autores y sus respectivas referencias cumplen con las pautas académicas.

Lugar y fecha:Lima, 23 de Septiembre del 2024.....

Apellidos y nombres del asesor / de la asesora: <u>Negrón Naldos, Luis Alfredo</u>	
DNI:10788917	Firma
ORCID: 0000-0003-1328-0323	

Resumen

El presente trabajo de consultoría se origina en respuesta a la problemática relacionada a la acumulación de deuda técnica acumulada en la empresa MS4M S.A.C. (en adelante MS4M), que se dedica al desarrollo y comercialización de productos tecnológicos para operaciones mineras a cielo abierto y subterráneas en Perú.

En los siguientes capítulos exploraremos la problemática, analizaremos como afrontarla, dimensionarla y gestionarla. Además, compartiremos buenas prácticas obtenidas de diferentes marcos de referencia, con el objetivo de fomentar un cambio cultural y estructural, a fin de alinearla a los objetivos de la empresa.

Inicialmente se llevó a cabo una evaluación interna que reveló una conexión significativa entre los problemas y el proceso del desarrollo de software. Esto afecta especialmente al componente de calidad de software, entre otros. Uno de los desafíos identificados es el escalamiento de soporte de nivel 1 al nivel 3, donde los desarrolladores senior atienden los problemas, en lugar de enfocarse en la mejora, desarrollo y evolución de los productos.

La implementación de los modelos propuestos descritos presenta numerosos desafíos provenientes de los marcos de referencias ágiles como DevSecOps, por lo que el presente trabajo intenta cubrir los problemas utilizando esta valiosa herramienta para evitar seguir acumulando una deuda técnica que incremente los costos de desarrollo y que le disminuya competitividad en el mercado.

De esta manera esperamos que el lector pueda comprender la importancia de la implementación de mecanismos de gobernanza, arquitectura y lineamientos de desarrollo de software, incluso en organizaciones que cuentan con productos posicionados en el mercado.

Abstract

This consulting work originates in response to the problem related to the accumulation of technical debt in the company MS4M S.A.C. (hereinafter MS4M), which is dedicated to the development and commercialization of technological products for open-pit and underground mining operations in Peru.

In the following chapters, we will explore the problem, analyze how to address it, dimension it, and manage it. Additionally, we will share best practices obtained from different reference frameworks, with the aim of promoting cultural and structural change to align with the company's objectives.

Initially, an internal evaluation was carried out, revealing a significant connection between the problems and the software development process. This particularly affects the software quality component, among others. One of the identified challenges is the escalation of support from level 1 to level 3, where senior developers address problems instead of focusing on improving, developing, and evolving products.

The implementation of the proposed models described presents numerous challenges arising from agile reference frameworks such as DevSecOps. Therefore, this work attempts to address the problems using this valuable tool to avoid further accumulation of technical debt that increases development costs and reduces market competitiveness.

In this way, we hope that the reader can understand the importance of implementing governance mechanisms, architecture, and software development guidelines, even in organizations that have established products in the market.

Tabla de Contenido

Resumen.....	3
Tabla de Contenido	5
Lista de Tablas	9
Lista de Figuras.....	12
Capítulo I: Situación general de la empresa	15
1.1. Historia de MS4M.....	16
1.2. Etapa inicial de desarrollo de productos	17
1.3. Etapa de inicio de comercialización.....	20
1.4. Etapa de crecimiento	21
1.5. Etapa de expansión.....	23
1.6. Productos, servicios y clientes	26
1.7. Visión, misión y valores de MS4M	28
1.8. Modelo de negocio de MS4M.....	30
Capítulo II: Análisis del contexto	34
2.1. Análisis de la industria	34
2.2. Análisis de la competitividad en la industria	36
2.3. Análisis externo.....	40
2.4. Análisis interno	42
Capítulo III: Metodología	47
3.1. Metodología de Trabajo	47
3.2. Identificación de Problemas	49
3.3. Determinación de Problemas principales.....	54
Capítulo IV: Revisión de la literatura	60

4.1.	Estado del Arte.....	60
4.1.1.	Sobre los marcos de referencia para el desarrollo de software	61
4.1.2.	Principios del desarrollo de software	62
4.1.3.	Metodologías del desarrollo de software	64
4.1.4.	Métodos de desarrollo de software	67
4.1.5.	Herramientas utilizadas	72
4.1.6.	Factores de éxito de los métodos de desarrollo de software	74
4.1.7.	Casos de éxito implementando métodos ágiles.....	77
4.2.	Casos Benchmarks relacionados	80
Capítulo V: Análisis del problema / Oportunidad detectada		85
5.1.	Causas identificadas	85
5.2.	Matriz de Priorización Causa-Raíz del Problema / Factibilidad de oportunidad detectada	92
Capítulo VI: Alternativas de solución evaluadas		98
6.1.	Alternativas para resolver el problema	98
6.1.1.	Implementar un modelo de gobernanza organizacional	98
6.1.2.	Implementar un modelo de arquitectura empresarial.....	103
6.1.3.	Implementar metodologías ágiles	106
6.1.4.	Subcontratar el desarrollo de software	107
6.1.5.	Incorporar desarrolladores de mayor experiencia	109
6.1.6.	Refactorizar el código de los productos	112
6.2.	Evaluación de alternativas.....	115
6.2.1.	Modelo de Evaluación	115
6.2.2.	Criterios de Evaluación	116

6.2.3. Resultados de Evaluación	132
6.2.4. Decisión Final	134
Capítulo VII: Solución Propuesta	136
7.1. Lineamientos, políticas base de gobierno sobre el desarrollo de software	136
7.2. Modelo de arquitectura empresarial para el desarrollo de software	144
7.3. DevSecOps como marco de referencia para el desarrollo de software	157
Capítulo VIII: Plan de Implementación y Factores clave de éxito	161
8.1. Actividades.....	161
8.2. Diagrama de Gantt de Implementación.....	162
8.2.1. Sobre los Lineamientos, políticas base de gobierno sobre el desarrollo de software	165
8.2.2. Sobre el Modelo de arquitectura empresarial para el desarrollo de software	165
8.2.3. Sobre DevSecOps como marco de referencia para el desarrollo de software.....	166
8.3. Presupuesto	168
8.4. Factores clave para el éxito	169
8.4.1. Adopción efectiva de metodologías ágiles.....	169
8.4.2. Implementación de políticas de gobernanza sobre el desarrollo de software.	169
8.4.3. Desarrollo de una arquitectura empresarial sólida.	169
8.4.4. Implementación exitosa de DevSecOps.....	170
Capítulo IX: Resultados esperados	171
9.1. Resultados esperados de la implementación	171
9.2. Resultados esperados en el frente social	177
Capítulo X: Conclusiones y recomendaciones	179
10.1. Conclusiones	179
10.2. Recomendaciones.....	180

Referencias.....	182
Anexo A: Ficha Técnica encuestas	198
Anexo B: Objetivos de corto plazo y tablero de mando de MS4M – Objetivo de largo plazo relacionado a las ventas	206
Anexo C: Objetivos de corto plazo y tablero de mando de MS4M – Objetivo de largo plazo relacionado a la rentabilidad	207
Anexo D: Estimación de costo promedio ponderado del capital en MS4M.....	208
Anexo E: Cálculo de Costo de Deuda en Nota 19a de Estado Financiero al 2023 de MS4M S.A.C.....	209



Lista de Tablas

Tabla 1 Crecimiento comercial y tecnológico de MS4M período 2014-2018.....	22
Tabla 2 Crecimiento comercial y tecnológico de MS4M período 2019-2023 (parte 1)	23
Tabla 3 Crecimiento comercial y tecnológico de MS4M período 2019-2023 (parte 2)	24
Tabla 4 Productos y servicios de MS4M	27
Tabla 5 Procedencia de ingresos de MS4M años 2021 - 2022	28
Tabla 6 Incidentes de soporte de sistema escalados años 2022	29
Tabla 7 Incidentes de soporte de sistema escalados años 2023	30
Tabla 8 Modelo de negocios de MS4M.....	31
Tabla 9 Análisis de la competitividad del sector de MS4M (parte 1)	36
Tabla 10 Análisis de la competitividad del sector de MS4M (parte 2)	37
Tabla 11 Matriz de evaluación de productos MS4M respecto a competidores	38
Tabla 12 Matriz de evaluación de productos MS4M respecto a competidores	39
Tabla 13 Matriz de evaluación de factores externos MEFE de MS4M	40
Tabla 14 Matriz de evaluación de factores internos MEFI de MS4M (parte 1)	42
Tabla 15 Matriz de evaluación de factores internos MEFI de MS4M (parte 2)	43
Tabla 16 Objetivos de largo plazo, hacia el año 2027 de MS4M.....	44
Tabla 17 Factores críticos de éxito para el mercado de MS4M.....	45
Tabla 18 Estado de resultados año 2022 de MS4M en dólares americanos	52
Tabla 19 Amenazas relacionadas al proceso de desarrollo de software	54
Tabla 20 Debilidades relacionadas al proceso de desarrollo de software.....	55
Tabla 21 Nivel de madurez de los procesos en MS4M	56
Tabla 22 Enfoque tradicional versus enfoque ágil.....	67
Tabla 23 Categorías de los factores críticos de éxito.....	76

Tabla 24 Ventajas y desventajas de métodos ágiles (parte 1).....	80
Tabla 25 Ventajas y desventajas de métodos ágiles (parte 2).....	81
Tabla 26 Prácticas de métodos ágiles más populares (parte 1).....	81
Tabla 27 Prácticas de métodos ágiles más populares (parte 2).....	82
Tabla 28 Practicas de métodos ágiles más populares	93
Tabla 29 Criterios para evaluar la influencia entre los problemas.....	95
Tabla 30 Experiencia y certificación de personal de investigación y desarrollo	110
Tabla 31 Comparación de alternativas de solución (parte 1).....	113
Tabla 32 Comparación de alternativas de solución (parte 2).....	114
Tabla 33 Escala de Juicios	115
Tabla 34 Comparación de criterios de Impacto de calidad de código (Ponderado)	116
Tabla 35 Normalización de criterio de Impacto de calidad de código	117
Tabla 36 Comparación de criterios: Impacto de complejidad y/o duplicidad (ponderado)...	118
Tabla 37 Normalización de criterio de Impacto de la complejidad y/o duplicidad	119
Tabla 38 Comparación de criterios de Impacto en la documentación (ponderado)	120
Tabla 39 Normalización de criterio de Impacto de la documentación	121
Tabla 40 Comparación de criterios de Impacto en la arquitectura (ponderado).....	122
Tabla 41 Normalización de criterio de Impacto en la arquitectura.....	123
Tabla 42 Comparación de criterios de Impacto en la tecnología obsoleta (ponderado)	124
Tabla 43 Normalización de criterio de Impacto en la tecnología obsoleta.....	125
Tabla 44 Comparación de criterios de Impacto en el control de calidad (ponderado)	126
Tabla 45 Normalización de criterio de Impacto en el control de calidad	127
Tabla 46 Comparación de criterios de Impacto en el despliegue (ponderado).....	128
Tabla 47 Normalización de criterio de Impacto en el despliegue.....	129

Tabla 48 Comparación de criterios Técnicos (ponderado)	130
Tabla 49 Normalización de criterios Técnicos	131
Tabla 50 Matriz de Priorización Criterios Técnicos	132
Tabla 51 Resultados de evaluación	134
Tabla 52 Probabilidad de riesgos	140
Tabla 53 Principios de la arquitectura	148
Tabla 54 Actividades de la solución propuesta (parte 1)	161
Tabla 55 Actividades de la solución propuesta (parte 2)	162
Tabla 56 Diagrama de Gantt (parte 1)	163
Tabla 57 Diagrama de Gantt (parte 2)	164
Tabla 58 Presupuesto	168
Tabla 59 Estimación de ahorros	172
Tabla 60 Otros Ingresos	173
Tabla 61 Evaluación económica expresada en soles	174
Tabla 62 Sensibilidad del factor de escalamiento a nivel 3	175

Lista de Figuras

Figura 1	Resumen de las etapas de la historia de MS4M	25
Figura 2	Organigrama de MS4M.....	26
Figura 3	Ventas anuales en soles de MS4M 2018-2022.....	50
Figura 4	Evolución de gasto de personal respecto a ingresos de MS4M 2018-2022	51
Figura 5	Evolución de la utilidad neta respecto a ingresos de MS4M 2018-2022	51
Figura 6	Horas de trabajo (%) del personal invertidas en soporte por mes (2023)	57
Figura 7	Horas de trabajo (%) del personal invertidas en soporte 2023, por producto	58
Figura 8	Secuencia de la metodología tradicional de cascada.....	65
Figura 9	Taxonomía de factores de éxito del desarrollo ágil.....	75
Figura 10	Síntesis de los factores críticos de éxito	77
Figura 11	Diagrama de causa y efecto de resultados de encuesta de percepción del personal sobre altos costos del soporte del nivel 3.....	86
Figura 12	Diagrama de proceso de soporte.....	89
Figura 13	Diagrama de Pareto de causas de incidencias relacionados al producto.	90
Figura 14	Diagrama de Pareto de causas de incidencias relacionados a la arquitectura	91
Figura 15	Diagrama de Pareto de causas de incidencias relacionados a calidad de código	91
Figura 16	Resultados de análisis causas-efecto aplicando Matriz de Priorización de Veste...96	
Figura 17	Resultados de priorización de causas-efecto en cuadrantes de Matriz de Vester....97	
Figura 18	Modelo de rendición de cuentas, retroalimentación y comunicación interna	100
Figura 19	Áreas de enfoque em el desarrollo de software.....	102
Figura 20	Beneficios de la implementación de arquitectura empresarial	104
Figura 21	Ciclo de vida del desarrollo de productos digitales	142
Figura 22	Método de desarrollo de arquitectura empresarial	144

Figura 23 Flujo de atención de requerimientos de desarrollo.....	146
Figura 24 Arquitectura de un producto digital en MS4M.....	149
Figura 25 Gobierno operativo federado.....	151
Figura 26 Proceso recomendado del ciclo de vida del desarrollo de un producto de software (Nivel 1).....	152
Figura 27 Proceso recomendado de análisis de requerimiento y planificación de productos de software (Nivel 2).....	152
Figura 28 Proceso de diseño de productos de software (Nivel 2).....	153
Figura 29 Proceso de implementación de productos de software (Nivel 2).....	154
Figura 30 Proceso de operación de productos de software (Nivel 2).....	155
Figura 31 Proceso de desuso de productos de software (Nivel 2).....	156
Figura 32 Flujo de integración y entrega continua.....	158
Figura 33 Sensibilidad del escalamiento proyectado versus el valor actual neto.....	176
Figura 34 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Calidad de código.....	199
Figura 35 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Falta de documentación.....	199
Figura 36 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a las Brechas de Seguridad.....	200
Figura 37 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Arquitectura Deficiente.....	200
Figura 38 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a Poca capacidad de Integración.....	201

Figura 39 Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto al Uso de Tecnología Obsoleta	201
Figura 40 Valoración del aspecto “Personal” sobre los tickets Nivel 3 en la atención de incidencias.....	202
Figura 41 Valoración del aspecto “Materiales” sobre los tickets Nivel 3 en la atención de incidencias.....	202
Figura 42 Valoración del aspecto “Indicadores” sobre los tickets Nivel 3 en la atención de incidencias.....	203
Figura 43 Valoración del aspecto “Mantenimiento” sobre los tickets Nivel 3 en la atención de incidencias.....	203
Figura 44 Valoración del aspecto “Madre Naturales” sobre los tickets Nivel 3 en la atención de incidencias.....	204
Figura 45 Valoración del aspecto al “Management” sobre los tickets Nivel 3 en la atención de incidencias.....	204
Figura 46 Valoración del aspecto “Respecto a la Metodología” sobre los tickets Nivel 3 en la atención de incidencias	205
Figura 47 Valoración del aspecto “Equipos y Maquinas” sobre los tickets Nivel 3 en la atención de incidencias	205

Capítulo I: Situación general de la empresa

En este capítulo se describirá a la empresa desde una perspectiva tecnológica, financiera y social, la situación general de MS4M, tanto interna como su interacción con el entorno, y se describirá el origen de la deuda técnica que posee y su repercusión financiera, así como las etapas de desarrollo de productos y modelo de negocio.

Es conveniente y necesario explicar, que, a juicio de los autores, es importante que se tome como referencia las siguientes líneas correspondientes a la historia de la empresa en estudio, como fuente donde se originan los problemas relacionados a la deuda técnica acumulada, que es el tema principal del presente trabajo.

Carver et al. (2022) definió a la deuda técnica como las actividades que no se han realizado en su debido momento y que ahora demandan una revisión por parte de los desarrolladores para cumplir con los requisitos no funcionales, esta deuda al final tiene los mismos componentes de una deuda financiera.

De acuerdo a lo indicado por Melo et al. (2022), esto puede deberse por fechas de entrega próximas, recursos limitados o la presión misma del mercado, no obstante, también puede deberse a un total desconocimiento por parte de los directores, gerentes o ejecutores de los proyectos acerca de la existencia de marcos de referencia, requerimientos no funcionales, o de la implicancia financiera en el largo plazo que ocasionará la deuda técnica que se acumulará año tras año, (Biazotto, et al., 2023).

Existen muchas estrategias disponibles para afrontarla, dimensionarla, cuantificarla y gestionarla (Melo et al., 2022), sin embargo, casi todas concurren que la necesidad de proveer la estructura para alinearla a una arquitectura objetiva, simplificación de interfaces de aplicación y el retiro de bases de datos y aplicaciones redundantes proveen un mayor impacto (Carver, et al., 2022).

El presente trabajo otorga buenas prácticas derivadas de marcos de referencia para evitar la deuda técnica en MS4M. Según la mayoría de los directores de las empresas que desarrollan tecnología, perciben que en los últimos tres años, la deuda técnica en las empresas que dirigen se ha incrementado, lo que ha reducido su capacidad para desarrollar nuevos productos y servicios (Carver, et al., 2022).

1.1. Historia de MS4M

MS4M es una empresa peruana que desarrolla y comercializa productos tecnológicos para operaciones mineras a cielo abierto y subterráneas en Perú, y que ha expandido su mercado hasta el extranjero desde la fecha de su fundación hace 10 años en Cajamarca. Su fundación se debe precisamente al emprendimiento de profesionales mineros con experiencia en sistemas de gestión de flotas en operaciones a cielo abierto en el Perú y en el extranjero. En el año 2013, año de su fundación, comenzó con 12 trabajadores incluyendo a los fundadores, Wilder Napoleón Pando Quevedo y Alejandro Ortiz Monteza, ambos ingenieros de minas egresados de la Pontificia Universidad Católica del Perú, con el objetivo de crear un sistema que comprende tanto software, hardware de procesamiento y almacenamiento local e infraestructura de bases de datos para almacenamiento a gran escala de los datos provenientes de los ciclos productivos de las operaciones unitarias en minería, con ingenieros de la ciudad de Cajamarca, Perú, lugar donde MS4M tiene su razón social incluso hasta el presente, llamándose inicialmente Mine Sense Solutions S.A.C. con un capital inicial de 240,000 dólares americanos.

Dividiremos para un mejor entendimiento del problema planteado, las etapas de la historia de MS4M acorde a los ciclos de vida de las organizaciones tecnológicas establecidas como base en diversos artículos técnicos donde se detalla las necesidades que abordan empresas que desarrollan tecnología desde su origen (Lindelöf & Hellberg, 2023).

1.2. Etapa inicial de desarrollo de productos

Los principios que se definieron para la elaboración de los productos iniciales fueron definidos en base empírica, sin principios de arquitectura específica, basado en la experiencia y conocimiento de los fundadores acerca de la codificación de algoritmos matemáticos y de circuitos electrónicos principalmente, con un equipo complementario conformado por programadores e ingenieros electrónicos, dentro de los cuales los criterios funcionales se priorizaron, sin considerar criterios no funcionales, a través de los siguientes lineamientos:

- Utilizar la amplia experiencia de optimización minera y programación de los socios.
- Usar y/o crear hardware que se encuentren al alcance de países del tercer mundo, que sea estable y fácil de reparar. Se definió incorporar a un ingeniero electrónico para tal fin.
- La tecnología de programación debió ser robusta y sencilla de aplicar, lo cual permitiría contratar personal de Cajamarca, reduciendo costos de traslado o incorporación de personal externo a la localidad, El sistema operativo seleccionado fue Android, a diferencia de la competencia principal que utiliza Windows CE.
- Para disminuir gastos, se decidió que las oficinas se ubicarían en la zona de Tres Molinos en Cajamarca. Esto permitiría contar con un ambiente de mayor concentración y amplitud para las pruebas de conectividad, además se evitaría la fuga de información. Las instalaciones también serían adecuadas para probar el hardware.
- Creación de un hardware capaz de soportar diferentes tipos de conexiones de equipos mineros, como Caterpillar, Komatsu, Hitachi, Volvo, etc.

- Usar una sola pantalla interactiva a ubicarse en la cabina de los operadores de flota para todas las soluciones tales como del sistema de administración de flota, monitoreo de presión y temperatura de llantas, monitoreo de uñas de equipos de carguío, sistema de fatiga de operadores y sistemas de anticolidión.

Estos criterios iniciales, fueron definidos de manera intuitiva y basándose de un conocimiento empírico del mercado, enfatizando que en las primeras etapas de los emprendimientos tecnológicos se prioriza un uso intensivo de recursos y una estructura organizacional simple, enfocada en las funcionabilidades y características de los productos a desarrollar (Lindelöf & Hellberg, 2023).

Sin embargo, Ligozat et al. (2022) planteó que antes del desarrollo de soluciones tecnológicas, es necesario evaluar el entorno, específicamente el mercado; principalmente la madurez de la tecnología que va a ser desplegada y la madurez del mercado donde el producto se va a introducir. Esta declaración de principios funcionales del producto inicial enfatiza implícitamente los niveles de incertidumbre que se advierte en la industria minera, y que se adhieren a cualquier emprendimiento tecnológico, por ello según explica su predilección por tecnología de un alto grado de madurez para sus procesos (Nwaila et al., 2022). En otras palabras, basándose en el entendimiento del negocio como su principal fortaleza, se tomaron decisiones de desarrollo de productos con un alto sentido de urgencia, asegurándose de que estuvieran alineadas con el mercado objetivo.

En esta primera etapa, se puso énfasis intensivo al entendimiento del negocio, que es necesario y crucial para el emprendimiento (Ligozat, et al., 2022). Esta comprensión requerida por parte del personal de desarrollo de productos le otorgaba una mejor idea de los requerimientos funcionales que se necesitaban desarrollar para adaptarse a las demandas de los futuros usuarios. Fue precisamente después de esta fase inicial cuando comenzó el diseño

de las estructuras lógicas a nivel conceptual, planteada por los especialistas del negocio, y no de arquitectos. Estas estructuras fueron entregadas a los desarrolladores para crear los primeros prototipos de bases de datos y variables de decisión. Este proceso de desarrollo, basado en la intuición, se inició sin considerar ningún requerimiento no funcional.

El desarrollo del hardware y software para la interfaz de los equipos a instalarse en las cabinas de los equipos mineros requirió capacidades de negocio no adquiridas hasta ese momento, por lo que se optó capacitar a los programadores en cursos por internet para que de forma simultánea puedan iniciar el diseño de las interfaces y las integraciones de las aplicaciones. Sin embargo, MS4M indicó en diferentes publicaciones del sector, que la etapa más desafiante fue la elección del hardware para adaptar e integrar a las soluciones creadas. Finalmente, se optó por una tableta comercial que fue sometida a todas las pruebas de compatibilidad y resistencia a vibraciones, simulando las condiciones de un equipo minero.

Superando la selección de componentes siguieron las actividades de sincronización entre el hardware, bases de datos, lógicas del sistema y otros dispositivos complementarios como el GPS. Estas actividades demostraron la poca precisión de las impresiones de las tarjetas que se realizaban localmente, por lo que los llevó a tomar la decisión de seleccionar a un proveedor en China para mejorar la calidad. Las pruebas funcionales fueron desarrolladas en las instalaciones del fundo Tres Molinos, en Cajamarca, hogar de uno de los fundadores, en dicha instalación se simularon circuitos mineros con vehículos de juguete y camionetas reales, en donde se ponían a prueba las funcionalidades desarrolladas y la sincronización con las lecturas del hardware instalado. Las etapas del diseño del hardware no contemplaron requerimientos no funcionales hasta la primera exportación llevada a cabo en el año 2018, tales como la hermeticidad, la seguridad intrínseca, radiaciones electromagnéticas, entre otras.

1.3. Etapa de inicio de comercialización

En el año 2014, MS4M es requerida sin proceso de licitación alguno, probar el prototipo del sistema de gestión de flota llamado por aquel entonces ControlSense, en la flota de producción directa de la mina a operarse por una empresa contratista. La empresa es Hudbay Perú S.A.C. en su mina Constancia, ubicado en la provincia de Cusco, a 4,100 metros sobre el nivel del mar, para una capacidad de procesamiento inicial de 80,000 toneladas por día mediante 19 camiones mineros de marca Caterpillar modelo 793F de 240 toneladas de capacidad, y 3 palas hidráulicas de marca Hitachi modelo EX5600 de 27 metros cúbicos de capacidad.

El acuerdo suscrito con Hudbay Perú consistía en una prueba piloto sin costo durante 6 meses, a raíz de la cual le permitía a MS4M demostrar las funcionalidades iniciales de su sistema y la robustez de su hardware. Esta primera versión comenzaba con prestaciones básicas de geo navegación asistida con precisión de 3 metros, conteo de viajes, cálculo de tonelaje transportado, medición de indicadores de utilización y de productividad. MS4M fue invitado a participar a inicios del 2014, hasta el mes de noviembre, fecha del despliegue e instalación, se llevaron a cabo procesos de validación y simulación de resultados, luego del cual se firmaría el contrato condicional al funcionamiento para la compra del sistema.

Este paso trascendental juega un papel muy importante en la historia de MS4M, en mercados con bajo nivel de madurez tal como lo representa una nueva operación minera sin antecedentes de sistematización anteriores, el enfoque sobre las funcionalidades es muy importante para el sostenimiento de las tecnologías a desplegarse, así como una efectiva comunicación de los beneficios y riesgos sobre los procesos a impactar (Ligozat, et al., 2022).

Considerando esta primera versión de ControlSense como una herramienta tecnológica con un bajo nivel de madurez o TRL (technological readiness level) cercano a 1,

en un mercado con bajo nivel de madurez, la incertidumbre relativa al funcionamiento y su entorno hace que se proyecte un crecimiento sobre mayores demandas a ser descubiertas mientras se son implementadas.

Seis meses después, MS4M pudo superar las pruebas de funcionalidad, convirtiendo a Hudbay Perú como su primer cliente, y ha mantenido esta relación hasta la fecha de elaboración del presente documento.

1.4. Etapa de crecimiento

Tan pronto como se estabilizó la solución implementada en Hudbay Perú, surgieron requerimientos adicionales, año tras año, las cuales se ven resumidas en los 3 años posteriores en la tabla 1, donde se muestran los contratos suscritos con nuevos clientes durante este período, así como nuevas funcionalidades incorporadas al producto: ControlSense como parte del requerimiento de sus próximos clientes.

Se puede observar que durante el período de 4 años comprendidos entre 2014 y 2018 se desarrollaron 16 módulos complementarios a la versión base lanzada en el 2013 lo que implicó una demanda de empleados para sostener este escalamiento de 12 a 75, consistentes principalmente en desarrolladores de software y de hardware, personal administrativo y ejecutivos de cuentas para gestionar las demandas.

Sin embargo, no sólo existió una demanda de evoluciones sino también de necesidad de soporte de primer nivel, segundo y tercero, sobre los productos ya desplegados en su etapa temprana de lanzamiento, comenzando así la historia que tendría que revisarse cuando el mercado le demande madurez de sus productos y sobre todo, competitividad en sus costos.

Tabla 1*Crecimiento comercial y tecnológico de MS4M periodo 2014-2018*

<i>Año</i>	<i>Empresa</i>	<i>País</i>	<i>Funcionalidades incorporadas</i>	<i>Cantidad de empleados</i>
2014	Hudbay Perú	Perú	Sistema ControlSense con GPS de baja precisión ReportSense.	12
2015	Buenaventura – Sociedad Minera El Brocal – Mina a cielo abierto	Perú	Sistema ControlSense con GPS de alta precisión. Modelo de bloques.	26
2016	Gold Fields La Cima – Cerro Corona	Perú	Modelo de compactación. Modelo de mezclas. Modelo de apilamiento para almacenamiento.	48
2017	Buenaventura – Sociedad Minera El Brocal – Mina subterránea	Perú	Nueva ControlScreen v1.0 Sistema de control de pesaje con balanzas en flota ligera. ControlSense para minas subterráneas	54
2018	Minsur – Marcobre	Perú	ControlScreen v2.0. Sistema antifatiga. Sistema de detección de proximidad Conexión a redes eLTE.	75
2018	Centerra Gold – Kumtor Gold Mine	Kirguistán		

Nota. Adaptado de Informe de gerencia general. Empresa MS4M. (2019). No publicado.

A medida que las empresas tecnológicas van superando sus primeros años, los recursos provenientes de sus primeras transacciones comerciales son materia de menor preocupación a diferencia de la nueva conciencia del entorno competitivo, donde los directivos mayormente se concentran en el corto plazo rechazando compartir ideas, problemas o soluciones en un sentido más amplio y de mayor alcance (Lindelöf & Hellberg, 2023).

Una de las razones de que las empresas incrementen su deuda técnica es precisamente por la falta de alineamiento de las políticas comerciales, la escasa definición de las capacidades de negocio o la vinculación de estas capacidades a la hoja de ruta de los productos o servicios (Carver, et al., 2022).

1.5. Etapa de expansión

Durante esta etapa, que incluye el escenario de pandemia por la COVID-19 (2020 – 2023), MS4M experimentó una etapa de crecimiento a nivel internacional en cuanto a clientes nuevos y sobre todo en la creación de nuevos productos dentro de su portafolio. Este crecimiento y demanda continúa se detalla en la tabla 2 y tabla 3.

Tabla 2

Crecimiento comercial y tecnológico de MS4M periodo 2019-2023 (parte 1)

<i>Año</i>	<i>Empresa</i>	<i>País debe tener la misma orientación las dos últimas líneas están alineadas a la derecha</i>	<i>Funcionalidades incorporadas</i>	<i>Cantidad de empleados</i>
2019	Buenaventura – Compañía Minera Coimolache.	Perú	ControlScreen v2.6 Sistema de monitoreo de neumáticos Sistema de detección de caída de elementos de desgaste en equipos de carguío.	140
2019	San Martín Contratistas Generales.	Perú	Sistema de producción con tonelaje de balanza	150
2020	Mansfield Minera – Mina Lindero.	Argentina	Sistema de gestión de atenciones a clientes. ControlSense para minas subterráneas v2.0. HealthSense. Módulo de pad de lixiviación. Reporte de control en tiempo real de equipos con alta precisión. Sistema de tracking para flota liviana. Sistema de alimentación eléctrica a través de puertos Ethernet (PoE).	
2021	Volcan Compañía Minera – Mina San Cristóbal.	Perú	ControlSense para minas subterráneas v3.0.	140

Nota. Adaptado de Informe de gerencia general. Empresa MS4M. (2023). No publicado.

Tabla 3

Crecimiento comercial y tecnológico de MS4M período 2019-2023 (parte 2)

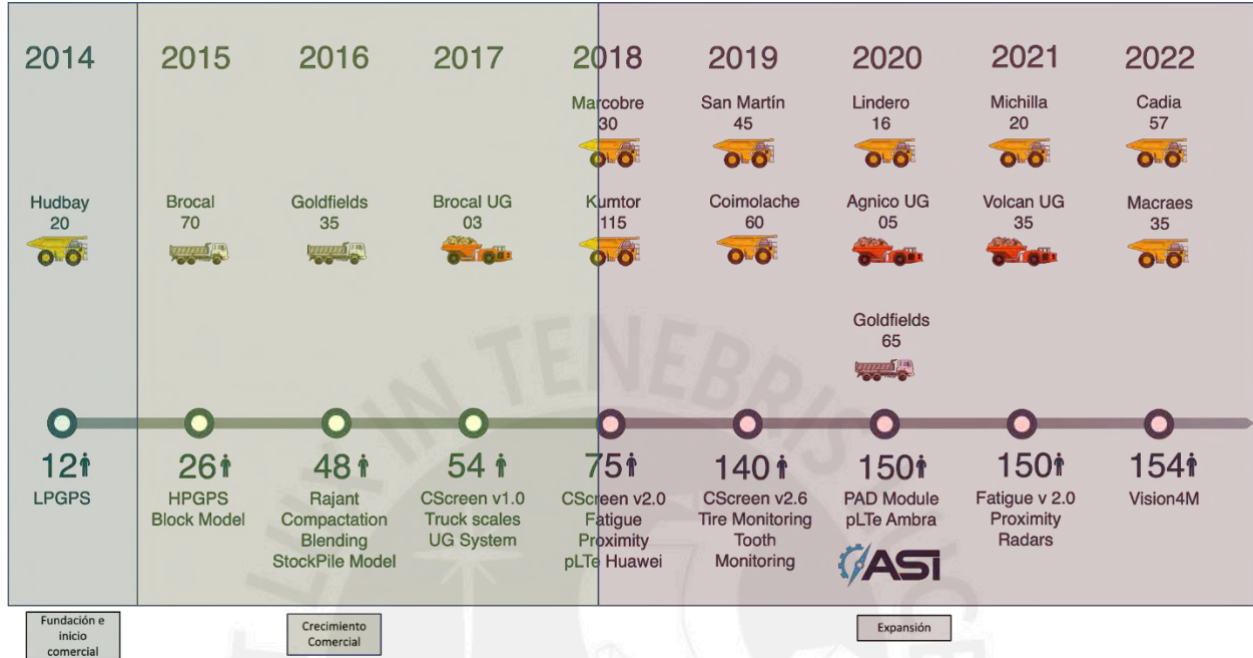
<i>Año</i>	<i>Empresa</i>	<i>País debe tener la misma orientación las dos últimas líneas están alineadas a la derecha</i>	<i>Funcionalidades incorporadas</i>	<i>Cantidad de empleados</i>
2021	MHC – Mina Michilla.	Chile	Sistema de detección de proximidad con bluetooth. Sistema antifatiga. Sistema de detección de proximidad con radares. Módulo de combustible Sistema de calidad de datos.	
2022	Oceana Gold – Macraes Mine.	Nueva Zelanda	Control 4Miners.	145
2022	Newcrest – Cadia Mine	Australia		
2023	Buenaventura – Sociedad Minera el Brocal – Mina subterránea.	Perú	Control 4Miners UG.	147

Nota. Adaptado de Informe de gerencia general. Empresa MS4M. (2023). No publicado.

En esta etapa MS4M estableció sociedades con oficinas en Chile, Australia, Canadá y Sudáfrica para obtener representación, desarrollo comercial y soporte local en sus nuevas cuentas en el extranjero. La Figura 1 muestra el resumen del crecimiento de MS4M desde el año 2013.

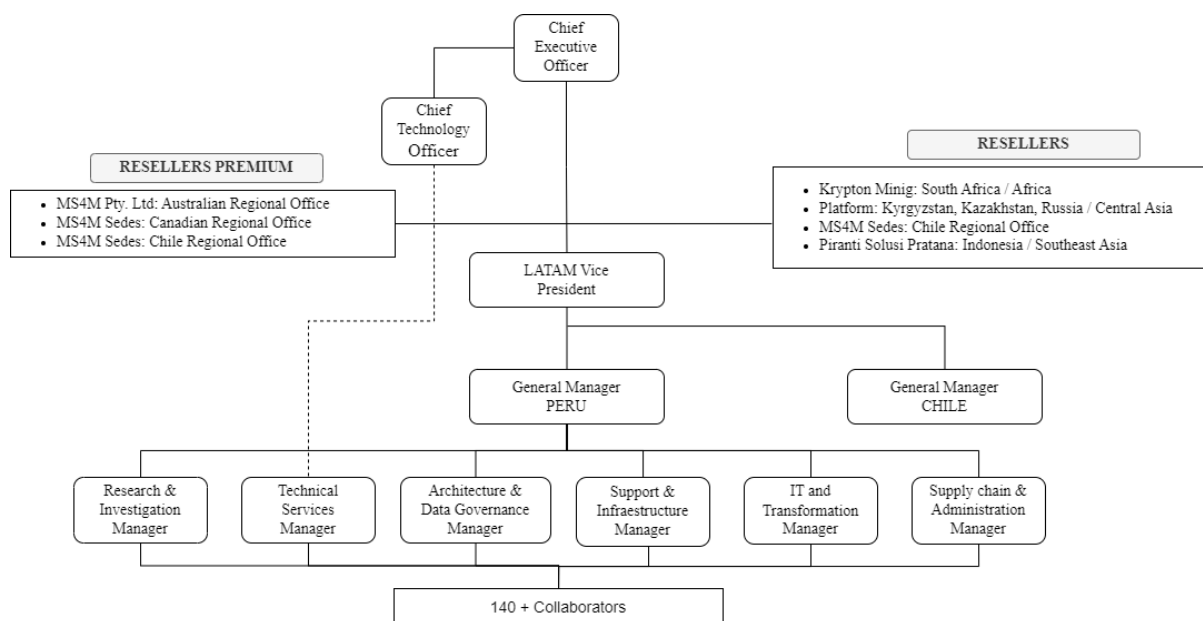
Figura 1

Resumen de las etapas de la historia de MS4M



Nota. Adaptado de Informe de Gestión de MS4M (2022) (No publicado).

En la siguiente Figura 2, se muestra el organigrama de MS4M y las capacidades de negocio, que tienen por objeto contribuir en la madurez de los procesos de la empresa, y aportar de manera positiva desafíos internacionales. Es preciso indicar que el primer arquitecto de software fue incorporado en la empresa 8 años después de creado a forma de consultoría externa, para ser incorporado a su línea gerencial a inicios del año 2022.

Figura 2*Organigrama de MS4M*

Nota. MS4M (2022). Diagrama organizacional de MS4M. Informe interno.

1.6. Productos, servicios y clientes

En la tabla 4 se muestra la cantidad de productos y servicios que conforman su cartera de oferta al sector minero.

Tabla 4*Productos y servicios de MS4M*

<i>Nombre</i>	<i>Descripción</i>	<i>Sector</i>
Control 4Miners	Sistema de gestión de flota para minería a cielo abierto, multimarca, con algoritmos de optimización y focalizado en el control operativo de la flota de producción minera directa, con capacidad de integración a otros sistemas de control mineros.	Minería a cielo abierto, metálica o no metálica
Control 4Miners UG	Sistema de gestión de flota para minería subterránea, multimarca, focalizado en el control operativo de la flota de producción minera directa y en la geolocalización en tiempo real de personas y equipos.	Minería subterránea
Report 4Miners	Sistema de inteligencia de negocios con dashboards. personalizados de carácter táctico y estratégico, focalizado en el tiempo real con base de datos SQL.	Todo tipo de minería
Health 4Miners	Sistema de gestión de activos multimarca, focalizado en el monitoreo de condiciones de salud de los activos mineros y con alertas configurables, multimarca.	Todo tipo de minería
FCS Fatigue Control System	Sistema de control de signos de somnolencia para operadores mineros con algoritmos de aprendizaje profundo para obtener datos en tiempo real.	Todo tipo de minería
PDS Proximity Detection System BT	Sistema de alerta y reporte de proximidades peligrosas entre la flota de producción minera, mediante el uso de bluetooth.	Todo tipo de minería
PDS Proximity Detection System Sensor	Sistema de alerta y reporte de proximidades peligrosas entre la flota de producción minera, mediante el uso de radares electromagnéticos.	Todo tipo de minería
TAS Tire Assessment System	Sistema de monitoreo de presión y temperatura de neumáticos.	Minería a cielo abierto
TPS Teeth Performance System	Sistema de monitoreo de caídas de elementos de desgaste de equipos de carguío, utilizando sensores y aprendizaje profundo.	Minería a cielo abierto
Servicio de diseño, implementación y soporte de redes inalámbricas para minería		Todo tipo de minería
Servicio de Optimización de flota	Servicio de operación del sistema Control 4Miners.	Todo tipo de minería
Servicio de administración de Control 4 Miners	Servicio de gestión del sistema Control 4Miners.	Todo tipo de minería
Servicio de soporte en campo del sistema Control 4Miners	Servicio de mantenimiento preventivo y correctivo de la infraestructura de Control 4Miners.	Todo tipo de minería
Servicio de soporte remoto de Control 4Miners	Servicio de corrección de bugs, firewalls, updates y upgrades.	Todo tipo de minería
Servicio de monitoreo de síntomas de fatiga	Servicio de monitoreo a través del sistema FCS Fatigue Control System en nivel de intervención temprana.	Todo tipo de minería

Nota. Adaptado de Informe de gestión Empresa MS4M. (2022). No publicado.

Gran parte de los ingresos también provienen de los servicios de soporte tecnológico que sus clientes reciben como complemento a la venta de los productos. MS4M ha optado por

ofrecer sus productos también en modalidad de alquiler, por lo que existe una marcada diferencia entre los ingresos proveniente de los servicios y de las ventas, lo que explica la orientación del mercado de no capitalizar activos tecnológicos por la incertidumbre global propia de la minería (Nwaila et al., 2022).

La relación entre estas dos fuentes de ingreso se muestra en la tabla 5 correspondiente a los años: 2021 y 2022.

Tabla 5

Procedencia de ingresos de MS4M años 2021 - 2022

<i>Año</i>	<i>Origen</i>	<i>Monto en dólares americanos</i>	<i>Porcentaje</i>
2021	Venta de productos	1,982,084	33.8%
2021	Servicios	3,874,661	66.2%
	Total 2021	5,856,745	100%
2022	Venta de productos	2,639,018	39.2%
2022	Servicios	4,087,687	60.8%
	Total 2022	6,726,705	100%

Nota. Adaptado de Estados financieros empresa MS4M (2021 - 2022). No publicado.

1.7. Visión, misión y valores de MS4M

MS4M ha declarado los principios rectores luego de un proceso de planeamiento estratégico llevado a cabo a finales del año 2021, a través de su plana gerencial, incluyendo directores en sesiones semanales durante 3 meses.

La visión de MS4M es: “Ser un referente mundial en soluciones de tecnología operacional minera”.

La misión de MS4M es: “Creamos soluciones innovadoras de gran valor para la industria minera, nuestro enfoque en la agilidad y calidad nos permite contribuir a la seguridad, productividad y sostenibilidad de las operaciones de nuestros clientes”.

En adición a lo indicado, se muestran a continuación, los valores señalados por MS4M: Enfoque en el cliente, Innovación, Agilidad, Calidad. Enfocando el análisis en la calidad de los productos, la deuda técnica prevalece como un indicador de esta. Así como la vigencia tecnología relacionada, y de todo el trabajo relacionado para actualizar los criterios de diseño y desarrollo de productos (Carver, et al., 2022). En la tabla 6 y tabla 7 se muestra la cantidad de incidentes de nivel 1 referidos principalmente a bugs o trabajos menores de configuración, en contraste con los incidentes escalados hacia el nivel 3 que requieren correcciones a nivel de código por los propios desarrolladores.

Tabla 6

Incidentes de soporte de sistema escalados años 2022

<i>Año</i>	<i>Origen</i>	<i>Atenciones de nivel 1</i>	<i>Atenciones de nivel 3</i>	<i>Escalamiento</i>
2022	Enero	235	132	36%
	Febrero	385	116	23%
	Marzo	681	132	16%
	Abril	569	112	16%
	Mayo	498	94	16%
	Junio	728	130	15%
	Julio	712	137	16%
	Agosto	592	175	23%
	Setiembre	615	115	16%
	Octubre	593	171	22%
	Noviembre	629	160	20%
	Diciembre	554	122	18%

Nota. Tomado de Informe de soporte tecnológico, Empresa MS4M. (marzo 2023). No

Publicado

Tabla 7*Incidentes de soporte de sistema escalados años 2023*

<i>Año</i>	<i>Origen</i>	<i>Atenciones de nivel 1</i>	<i>Atenciones de nivel 3</i>	<i>Escalamiento</i>
2023	Enero	673	134	17%
	Febrero	882	160	15%
	Marzo	980	254	21%

Nota. Tomado de Informe de soporte tecnológico, Empresa MS4M. (marzo 2023). No

Publicado

Con un nivel de escalabilidad de 20% en promedio, significa que el equipo de investigación y desarrollo I+D se mantiene activo en la atención de incidentes del nivel 3, este último definido por la necesidad de realizar modificaciones al código fuente de la solución, y con ello dar soporte a los productos de MS4M, esta inversión de tiempo y esfuerzo del personal tiene un efecto financiero y técnico que se detallará más adelante.

1.8. Modelo de negocio de MS4M

Es necesaria una integración de las necesidades, junto con la definición del modelo de negocios establecidas por MS4M en el lienzo de canvas, precisamente resaltando el beneficio del enlace de la perspectiva futura tecnológica junto con la realidad actual (Nazarenko, et al., 2022).

A través de la descripción de la tabla 8, se detalla el modelo de negocio de MS4M, el cual considera los factores del Business Model Canvas.

Tabla 8*Modelo de negocios de MS4M*

<i>Elemento</i>	<i>Descripción</i>
Propuesta de valor	<ul style="list-style-type: none"> • Gestión integral de la flota de producción en mina. • Optimización de los ciclos de minado. • Control de la dilución de minado. • Gestión de la vida de los activos mineros de producción directa. • Prevención de accidentes por colisiones entre equipos y por fatiga de operadores. • Control de envío de inchancables a la planta. • Datos confiables, en tiempo real del resultado del proceso de minado. • Un fuerte, confiable y efectivo soporte tecnológico. • Un sólido enfoque consultor.
Relación con clientes	<ul style="list-style-type: none"> • Equipo de profesionales mineros con alto conocimiento de gestión de flota dedicado a la gestión de requerimientos en mina. 24/7.
Canales de distribución	<ul style="list-style-type: none"> • Marketing digital. • Red de contactos empresariales.
Segmento de clientes	<ul style="list-style-type: none"> • Medianas y grandes empresas mineras a cielo abierto y subterráneas, metálicas y no metálicas, en el Perú y en el extranjero. • Greenfields tienen una especial atención en la medida de construir relaciones a largo plazo. • Brownfields requieren de atención focalizada en las oportunidades de negocio, en el corto y mediano plazo.
Actividades clave	<ul style="list-style-type: none"> • Desarrollo y comercialización de sistemas de gestión de flota de producción minera, gestión de activos operacionales mineros y de control operacional, para operaciones mineras a cielo abierto y subterráneas. • Soporte tecnológico. • Atención al cliente.
Recursos clave	<ul style="list-style-type: none"> • Personal de I&D multi disciplinario, con conocimiento de negocio. • Hardware y Software certificado y patentado. • Oficinas regionales.
Asociados clave	<ul style="list-style-type: none"> • Proveedores de circuitos electrónicos. • Alianzas con socios de tecnologías líderes en el mercado de telecomunicaciones. • Asociaciones gubernamentales de financiamiento a emprendimientos o proyectos tecnológicos.
Estructura de costos	<ul style="list-style-type: none"> • Remuneraciones. • Costos de investigación y desarrollo de productos. • Equipos informáticos. • Licencias de Software. • Infraestructura (SAAS; PAAS).
Flujo de ingresos	<ul style="list-style-type: none"> • Venta de sistemas e infraestructura de red. • Renta de sistemas e infraestructura de red. • Consultorías técnicas operacionales. • Servicios de optimización de flota. • Servicios de soporte en mina y remoto.

Nota. Tomado de Informe de mejora de procesos. Empresa MS4M. (2022). No publicado.

MS4M se dedica a desarrollar soluciones tecnológicas para la optimización y control operacional de la industria minera, a través del desarrollo de dispositivos electrónicos y algoritmos que optimizan el control de la operación de sus clientes.

La empresa cuenta con 160 colaboradores donde son los profesionales mineros los que predominan en términos de su relación con los clientes, enfatizando el conocimiento del negocio y de sus necesidades. Su mercado no distingue entre tamaños de empresas mineras usualmente clasificadas por sus volúmenes de producción, pues sus productos son aplicables a todos los segmentos de mercado de la minería, e incluso, las modalidades de colocación pueden ser de venta directa o de alquiler, facilitando su accesibilidad. De igual forma, mantiene recursos humanos especializados en el sector. Sus principales costos se concentran en las planillas y en el desarrollo de nuevas soluciones.

Considerando esto como factor importante, un alto costo en el desarrollo de software implica que la deuda técnica puede encarecer los trabajos directos para la creación de nuevos productos al extender el tiempo de salida al mercado (time to market) y afectar la retención del personal estratégico ya que el objetivo de los desarrolladores es de manera preferencial crear nuevos productos antes que brindar soporte. Es por esta razón que este modelo debe ser escalado, considerando que la principal fuente de ingresos proviene de servicios que incluyen el alquiler del sistema Control 4Miners y del soporte tecnológico en la mina y no de las ventas de los productos desarrollados.

El escalamiento de su actual modelo de negocios le permitiría alcanzar los objetivos de largo plazo descritos en la tabla 16, donde se incorporaría a la propuesta de valor, soluciones integradas con inteligencia artificial, acorde a las nuevas tendencias de demanda de productos tecnológicos para los próximos años (Buchholz, et al., 2021) donde se considera la incorporación de capacidades internas proveniente de sus procesos y habilidades de su

personal, en el marco de un escenario posterior a la pandemia de la COVID-19 que le permitiría obtener una mayor capacidad de retención del talento (Abenov, et al., 2023).



Capítulo II: Análisis del contexto

En este capítulo se describirá el entorno externo de MS4M, evaluando a la industria y los costos asociados con el desarrollo de los productos y sus funcionalidades, así como la comercialización y los insumos (importaciones) requeridos para su fabricación, considerando también el tiempo necesario para adquirir estos insumos y su impacto en MS4M.

2.1. Análisis de la industria

El sector minero tiene tradicionalmente una valoración referente al precio de los metales, la calidad de su mineral, ubicación, confiabilidad de su equipo y activos físicos, e invierte una considerable cantidad de tiempo y recursos para optimizar su producción, a un 2% (Abenov, et al., 2023). En este sentido, la empresa en análisis y sus competidores en aras de ofrecer un mejor tiempo de respuesta, fiabilidad y soporte efectivo, deben considerar la implementación de marcos de referencia para la gestión interna del proceso de desarrollo que requieren sus productos, fortaleciendo la calidad percibida ante los clientes a través de procesos de desarrollo alineados a buenas prácticas.

Esta industria se ve altamente demandada por altos costos de requerimientos de infraestructura operativa y energética, horizontes muy distantes de planificación y regulación estatal, desenvolviéndose en entornos operativos de alto riesgo operativo con muchas demandas por controles de seguridad (Minería en Línea, 2023), debido a esto, cada uno de los equipos (antenas, sensores, vehículos solares, paneles, otros) implementados en las operaciones mineras, presentan riesgos físicos a lo largo de la operación minera, por consecuente, los controles de seguridad de estos, deben contar con enfoque de gestión preventiva, proactiva y reactiva, buscando que el impacto ante eventos negativos no controlados, sean minimizados y/o eliminados en la operación.

En ese orden, la industria minera está constantemente puesta a prueba, asumiendo retos constantes y coyunturales tales como la presencia de mayor cantidad de minas antiguas y más profundas que incrementan los gastos asociados al transporte, muchos riesgos relacionados a la propiedad de los terrenos, acceso a infraestructura de calidad por su lejanía, y creciente demanda nacionalista de los recursos, en este contexto la búsqueda permanente de mejora en los procesos recae en la tecnología, la cual se utiliza como una herramienta para diversos objetivos, entre los cuales destacan la mejora de eficiencia en procesos de mayor costo, generar información confiable para mejor toma de decisiones de manera holística e integrada, con la capacidad de predecir resultados y activar acciones correspondientes (Feng, et al., 2022).

La tecnología que contribuirá estos objetivos se inclinará por la automatización de la flota para prescindir de los operadores e incrementar la productividad, el internet de las cosas, para obtener información y activar acciones automáticas en tiempo real de los activos mineros y de esa forma incrementar su vida y su disponibilidad mecánica, la inteligencia artificial que permitirá analizar grandes cantidades de datos e identificar patrones y tendencias de forma rápida y confiable, la realidad virtual y aumentada para disminuir el riesgo de entrenamiento, y el uso de energías renovables para disminuir la dependencia energética de fuentes contaminantes, contribuyendo a la sostenibilidad del negocio minero (Swart, et al., 2021). En este sentido, dada el alto volumen de datos que son generados y capturados por los equipos de monitoreo instalados en los activos de la operación, se considera inherente la aplicación y construcción de herramientas basadas en la inteligencia artificial, esto aunado a los estándares vigentes, multiplicará de manera exponencial el valor de los productos ofrecidos, y en consecuencia la brecha de competitividad será cada vez mayor para las empresas que no adopten dichas tecnologías emergentes.

2.2. Análisis de la competitividad en la industria

Basado en el mismo proceso estratégico desarrollado, MS4M siguiendo a (Laluf, 2022), realizó el análisis de la competitividad en el sector. La tabla 9 y tabla 10 resume los 5 factores que recomienda la metodología.

Tabla 9

Análisis de la competitividad del sector de MS4M (parte 1)

<i>Elemento</i>	<i>Descripción</i>
Amenaza de nuevos ingresantes	<ul style="list-style-type: none"> • La amenaza es moderado -elevado ya que varias empresas de tecnología pueden incursionar en el sector de MS4M. SAP podría entrar a la industria (en el mediano plazo puede comprar un sistema de gestión de flota; también pueden entrar los desarrollos in house, softwares brasileiros de bajo costo y otros similares. • Varias empresas están invirtiendo para ingresar a nuestro mercado, existen asociaciones, alianzas. • Aparecen nuevos actores que son complementos de nuestros grandes competidores (ejemplo Modular posiblemente adquiere a Octopus). • Además, empresas pequeñas también están apareciendo y pueden convertirse en una amenaza. • Amenaza por posibles adquisiciones de empresas grandes. • Posible entrada de empresas chinas, (bajo costo).
Poder de negociación de los compradores	<ul style="list-style-type: none"> • El poder de los compradores (clientes) es moderado. Los clientes de MS4M a pesar de ser grandes empresas mineras, nuestros productos son de primer nivel y diferenciados. • Por la inversión que significa y por la existencia de algunos competidores, los clientes son más detallistas con los requisitos. Poderes que se deben de cumplir en una licitación, y delimitan el alcance para cumplir con estándares internacionales y certificaciones. • Las grandes mineras quieren soluciones probadas no arriesgan en pilotos. • La mediana minería no piensa mucho en implementar estas soluciones porque son muy caras y ahí tenemos una oportunidad, también podemos ofrecer soluciones donde paguen por lo que consuman. • El poder de los compradores es más grande en telecomunicaciones porque tiene más opciones en el mercado. • En el caso de sistemas de gestión de flota no lo es tanto, depende mucho del tamaño de la mina, como Antamina que tiene mayor poder por su tamaño, pero hay otras minas de menor envergadura donde el poder del comprador es menor. • Es importante señalar que minas como Antamina no hay muchas. • Si nos metemos en la columna vertebral de los procesos del cliente, ejercemos un poder ya que cambiar el sistema es muy difícil. • Las primeras cuentas fueron apuestas, se tienen muchos contactos en minería. • Se han generado compromisos que hacen ver que el poder está más del lado de los clientes. En el caso de clientes por licitaciones, la relación comercial es diferente. • En explosivos te pueden cambiar. En nuestro caso es más difícil. En nuestro caso podemos controlar el poder de nuestros clientes. • Nosotros tenemos un poder de conocimiento (know how minero) e información.

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022).

No publicado.

Tabla 10*Análisis de la competitividad del sector de MS4M (parte 2)*

<i>Elemento</i>	<i>Descripción</i>
Amenaza de los productos sustitutos	<ul style="list-style-type: none"> • El poder de los sustitutos es moderado por la flota autónoma. • Por ejemplo, un sustituto es la flota autónoma. • No han aparecido muchos sustitutos en el mercado. • Actualmente existe tecnología open source y los componentes electrónicos son más fáciles de conseguir. • Siempre existe el riesgo que nos copien nuestras soluciones. • FMS de bajo costo puede ser un producto sustituto. (o de menor precio que nosotros). • Algunos productos existentes no llegan a cubrir todas las necesidades de nuestros clientes. • En la pequeña minera los productos de las empresas de bajo costo podrían ser sustitutos. • En la mediana o gran minería, va más por la flota autónoma. • Red de contactos empresariales.
Poder de negociación de los proveedores	<ul style="list-style-type: none"> • El poder de los proveedores es moderado a elevado. • Existe escasez de productos. • En la medida que se proyecten cantidades el poder puede ir bajando. • No tenemos poder de negociación por volumen. • No contamos con componentes CORE técnicamente sustituibles debido al diseño de producto. • Se debe trabajar sobre esto, no tenemos poder de negociación porque todavía nuestras compras son por proyectos, tal vez falta el financiamiento para comprometer forecast de productos y no tener quiebres de stock, esto a veces, nos juega en contra en nuestros compromisos. • Los proveedores de procesadores ejercen un poder elevado porque son difíciles de cambiar.
Rivalidad de los competidores	<ul style="list-style-type: none"> • La rivalidad es baja. • La competencia es medianamente diferenciada. • No hay muchos competidores (en las licitaciones no hay muchos competidores). • Las barreras de salida son bajas, las barreras de entrada son un poco elevadas. • Es posible que un competidor nos copie, pero el riesgo es mínimo.

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No publicado.

Sin embargo, se sugiere que la competitividad del sector tecnológico sea cual fuere la industria, tenga en consideración los factores como: (1) eventos mundiales y geopolíticos, (2) la expectativa y demanda de los clientes y usuarios ante una mayor actividad en redes y expectativas para que la tecnología genere valor en sus procesos, (3) las tecnologías emergentes y tendencias, (4) la dinámica de la industria, (5) los nuevos ingresantes, nuevas alianzas, (6) los modelos de negocios. (Minería en Línea, 2023). En este sentido, la aceptación de estos factores en la industria, en los competidores y en la propuesta de valor,

favorecerá el desarrollo de propuestas de valor con enfoques transversales a la industria, representando retos en cada uno de los actores del mercado, y los cuales, dependiendo del tamaño de la organización, la cantidad de productos y el número de clientes, favorecerá su integración a los procesos de desarrollo de productos.

Por consiguiente, como parte del estado situacional se identificó como propuesta, que la empresa MS4M evalúe la competitividad de su sector con enfoques actualizados y determinar su situación frente al mercado.

Desde un punto de vista descriptivo, se llevó a cabo un análisis de los principales productos disponibles para el mercado objetivo, el cual se muestra en la siguiente tabla 11 y tabla 12.

Tabla 11

Matriz de evaluación de productos MS4M respecto a competidores

<i>Oportunidades</i>	<i>Competidores</i>	<i>Características</i>	<i>MS4M</i>	<i>Estrategia</i>
Control 4Miners	Modular	Longevo, reputación	Mas eficiente	Diferenciador
	Hexagon	Reputación	Mas económico	Costos
	Caterpillar	Mediana reputación, sinergia	Mas eficiente, económico	D&C
	Wenco	Opción económica	Mas eficiente	Diferenciador
Report 4Miners	Modular	No tiene, usa Power BI	Mas eficiente	Diferenciador
	Hexagon	J View limitado	Mas eficiente	Diferenciador
	Caterpillar	No tiene	Mas eficiente	Diferenciador
	Wenco	AVOCA Limitado	Mas eficiente	Diferenciador
Health 4Miners	Modular	Eficiente, relativamente caro	Mas eficiente, económico, sinergia	Diferenciador
	Hexagon	No tiene	Mas eficiente, económico, sinergia	Diferenciador
	Caterpillar	Eficiente, relativamente caro	Mas eficiente, económico, sinergia	Diferenciador
	Wenco	No tiene	Mas eficiente, económico, sinergia	

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No publicado.

Tabla 12

Matriz de evaluación de productos MS4M respecto a competidores

<i>Oportunidades</i>	<i>Competidores</i>	<i>Características</i>	<i>MS4M</i>	<i>Estrategia</i>
FCS Fatigue Control System	Hexagon	Relativamente económico, posicionado Eficiente, relativamente caro	Mas eficiente, económico, sinergia	D&C
	Caterpillar		Mas eficiente, económico, sinergia	D&C
PDS Proximity Detection System.	Hexagon	Posicionado	Mas eficiente, económico, sinergia	D&C
	Torsa	Eficiente, caro	Mas eficiente, económico, sinergia	D&C
TPS Teeth Performance System	Motion Metrics	Caro, eficiente, buena reputación	Mas eficiente, económico, sinergia, flexible	Costos

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No publicado.

Respecto a los productos, se estima que dentro de las tecnologías estratégicas que ayudarán a las empresas a afrontar los retos de su entorno, se encuentran lo que se denomina observación aplicada, que consiste principalmente en obtener la mayor cantidad de sonorización en tiempo real para tomar decisiones de forma autónoma, usando a su vez inteligencia artificial de forma tal que se las organizaciones se vuelvan basadas en datos (Groombridge, 2023). Quien afirma también que la inteligencia artificial deberá ser adaptable, permitiendo un desarrollo e implementación rápida para las operaciones. Esta adaptación permitirá reducir los tiempos de entrega, estimándose una mejora del 25 % en comparación con los modelos de inteligencia artificial que no lo sean. Esto conllevaría a orientar el desarrollo de los productos de MS4M hacia esa línea, teniendo claro sus valores de diferenciación respecto a sus competidores.

2.3. Análisis externo

Durante el mismo proceso estratégico, MS4M identificó las oportunidades y amenazas propias en su entorno, usando el proceso estratégico y un análisis PESTEL (Laluf, 2022), cuyo resumen se muestra en la tabla 13.

Tabla 13

Matriz de evaluación de factores externos MEFE de MS4M

<i>Oportunidades</i>	<i>Peso</i>	<i>Valor</i>	<i>Puntaje</i>
Preferencia-tendencia a menores tiempos de entrega por parte de los clientes.	4%	3	0.13
Alta demanda de sistemas de fácil y rápida instalación, escalables, con capacidad de ejecutar migraciones desde otros sistemas con el menor impacto en los procesos del cliente.	6%	3	0.19
Alta demanda de tecnología por la transformación digital en operaciones subterráneas, OP en Chile.	6%	2	0.11
Cientes insatisfechos en el mercado.	6%	3	0.19
Ambiente económico favorable por el tipo de cambio e incremento de los precios de los metales.	4%	2	0.08
Existencia de otros mercados no atendidos que son atractivos.	5%	2	0.11
Acuerdos bilaterales y TLC con diversos países.	4%	2	0.08
Existencia de fondos concursables para el financiamiento de proyectos con asociación a ecosistema de startups.	4%	1	0.04
No existe una red económica para UG.	6%	1	0.06
Cientes demandan alta capacidad de integración a sus sistemas que son proporcionados por sus proveedores.	5%	2	0.11
<i>Amenazas</i>	<i>Peso</i>	<i>Valor</i>	<i>Puntaje</i>
Creciente presencia de competidores de bajo costo brasileros y otros.	5%	2	0.091
Existencia de restricciones-barreras por ser de origen peruano/ latinos.	5%	2	0.108
Existencia de restricciones OEM para flota autónoma y tele operación.	5%	2	0.098
Fuga de conocimientos y/o talentos.	7%	2	0.136
Regulación de países no permitan la exportación de componentes electrónicos para I&D.	4%	1	0.035
Existe alta demanda de productos y servicios certificados y con propiedad intelectual a nivel internacional.	5%	1	0.046
Escasez de componentes electrónicos a nivel mundial.	5%	2	0.100
Proveedores internacionales no confiables.	4%	2	0.085
Componentes descontinuados en el mercado con mayor rapidez.	5%	1	0.054
Competidores comienzan a adoptar nuevas tecnologías.	5%	2	0.100

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No

publicado.

La valoración 4 otorga alta capacidad para responder, valoración 3 otorga una capacidad normal, valoración 2 otorga una capacidad débil y la valoración 1 otorga una capacidad con un muy bajo nivel de respuesta. Analizando el resultado de esta evaluación, dentro de las principales oportunidades que se concentran en el mercado y sus preferencias, donde la demanda de agilidad es implícita, pero, sobre todo, plasmada en tiempos de entrega y productos flexibles que determinan clientes insatisfechos con sus proveedores actuales. La metodología usada para esta clasificación (Laluf, 2022) demanda la ejecución de estrategias para aprovechar las oportunidades y neutralizar las amenazas, donde prevalecen criterios de calidad por el origen latino, posible fuga de talentos y la crisis de disponibilidad de procesadores a nivel mundial. Respecto a esto, se estima que 56% de los CIO espera implementar metodologías ágiles de entrega flexible o progresiva que le permita incrementar la capacidad de respuesta para de esa forma tener un mayor estímulo financiero interno (Buchholz, et al., 2021). MS4M debe considerar la adopción de un marco de referencia para ese propósito.

Advierte también un considerable riesgo de amenaza para la retención de talentos, (Swart, et al., 2021) asegura también que los trabajos en minería, continúan siendo una actividad todavía primaria en términos industriales, y que se han transformado debido al escenario posterior a la pandemia, donde las habilidades digitales de los profesionales en minería siguen siendo transformados hacia menos fundamentos, pero más habilidades, proporcionando al sector de MS4M una mayor cantidad de talentos a reclutar con enfoques distintos. Sin embargo, para la retención del talento, existen muchas estrategias asociadas tales como “tecnología para la tecnología” como lo advierte (Buchholz, et al., 2021) donde los directores empresariales deben facilitar la creación de tecnología invirtiendo en mejores

prácticas o en infraestructura para no sólo apalancar financieramente el emprendimiento tecnológico si no la generación de una cultura basada en el producto, de no culpabilidad y de esfuerzo por entregas funcionales progresivas basadas en metodologías ágiles.

2.4. Análisis interno

Durante el mismo proceso estratégico, MS4M identificó las fortalezas y debilidades, concentradas desde un análisis AMHOFIT previo, usando el proceso estratégico (Laluf, 2022). La evaluación considera el peso de fortaleza identificada por el valor asignado por el equipo, obteniendo el puntaje mostrado en la tabla 14 y tabla 15.

Tabla 14

Matriz de evaluación de factores internos MEFI de MS4M (parte 1)

<i>Fortalezas</i>	<i>Peso</i>	<i>Valor</i>	<i>Puntaje</i>
Conocimiento y experiencia en el negocio minero.	7.5%	4	0.30
Credibilidad en nuestra capacidad para satisfacer las necesidades de los clientes	6.3%	4	0.25
Agilidad para atender las necesidades de los clientes	5.8%	4	0.23
Equipo altamente comprometido (motivado) y autodidacta.	5.5%	4	0.22
Ecosistema de soluciones integrado al estándar operacional (Internacional)	4.8%	3	0.14
Desarrollamos nuestra propia tecnología y gestionamos su evolución. (Desarrollo, evolución y mejora continua)	5.4%	4	0.22
Capacidad para captación y formación de jóvenes profesionales.	4.1%	3	0.12
Alta capacidad de despliegue con el mínimo de personal y de manera remota.	4.5%	3	0.14
Nuestras propuestas, productos y servicios comerciales son escalables.	4.8%	3	0.14

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M (2022). No publicado.

Tabla 15*Matriz de evaluación de factores internos MEFI de MS4M (parte 2)*

<i>Debilidades</i>	<i>Peso</i>	<i>Valor</i>	<i>Puntaje</i>
Procesos y procedimientos internos no están definidos (No hay ERP, en logística, etapas de proyectos)	6.9%	1	0.07
No se cuenta con planes ni presupuestos para compras, ventas, contrataciones, etc. (Gestión presupuestal)	6.3%	1	0.06
Estructura organizacional no es escalable.	5.1%	2	0.10
Falta de la gestión del conocimiento (una parte es la documentación técnica) y comunicación (los que hay no están socializados)	4.9%	2	0.10
Poco control financiero e integral de los proyectos.	4.9%	1	0.05
Alta dependencia de proveedores electrónicos (productos terminados con materiales/componentes poco compatibles-insustituibles)	4.8%	1	0.05
Débil gestión de QA/QC	4.5%	1	0.05
Poca capacidad de retención de talentos y Falta de gestión de personas	5.0%	1	0.05
Productos en etapa de no madurez (“evolución”) que ocasiona una alta demanda de soporte.	4.9%	1	0.05
No hay etapas definidas para los productos (ciclo de desarrollo y ciclo de vida)	4.4%	2	0.09

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M (2022). No publicado.

La valorización corresponde a 4 como una fortaleza mayor, 3 a una fortaleza menor, 2 a una debilidad menor y 1 a una debilidad mayor. Este análisis muestra que MS4M debe prestar mayor atención a las debilidades ya que son mayores que sus fortalezas, demostrando que tiene una posición interna ligeramente débil ya que obtuvo un puntaje de 2.42 donde la media es 2.5. Centrando el análisis en las debilidades identificadas en MS4M, existen fuertes problemas estructurales asociados a procesos, estructura organizacional y planificación. Se comprende a partir de esta evaluación la necesidad de emprender un proceso de transformación interna, donde se estandaricen procesos definidos en una arquitectura empresarial que incluya la gestión sobre las operaciones y procesos, la asociación con clientes y proveedores, la creación y desarrollo de servicios, y la gestión del talento y recursos en general, para que le permita afrontar las necesidades y demanda actual del mercado de tecnología (Minería en Línea, 2023).

Como resultado final de estas evaluaciones, MS4M definió sus objetivos estratégicos de largo plazo hacia el 2027, producto de la necesidad de abordar al mercado en el contexto posterior a la pandemia. Los objetivos de largo se muestran en la tabla 16.

Tabla 16

Objetivos de largo plazo, hacia el año 2027 de MS4M

<i>Perspectiva</i>	<i>Resultado estratégico</i>	<i>Objetivos de largo plazo</i>
Financiera	Accionistas Satisfechos	<ul style="list-style-type: none"> • MS4M al 2027, tendrá ingresos por ventas no menores a 12.5 millones de USD al año. • MS4M al 2027 generará un EBITDA de 40% • MS4M al 2027 tendrá una participación de mercado de 10% en cada continente.
Clientes	Clientes satisfechos	<ul style="list-style-type: none"> • MS4M al 2027, será capaz de brindar soporte a sus clientes con un nivel de satisfacción mayor al 90%, esto representará una reducción del 80% de los casos de nivel 2 y 3 respecto del 2022.
Procesos	Procesos eficientes	<ul style="list-style-type: none"> • MS4M al 2027, tendrá conectividad con el 100% de marcas de equipos mineros. • MS4M al 2027, tendrá al menos una operación minera en cada continente con soluciones autónomas o inteligencia artificial en gestión de flota. • MS4M al 2027, tendrá el 100% sus productos y servicios con patentes y certificaciones internacionales aplicables.
Aprendizaje	Empleados motivados y preparados	<ul style="list-style-type: none"> • MS4M al 2027, será reconocida como GREAT PLACE TO WORK • MS4M al 2027, tendrá sistematizados y certificados el 100% sus procesos internos en su cadena de valor

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No publicado.

MS4M ha identificado los factores de críticos de éxito durante su planeamiento estratégico iniciado en octubre del 2021 y finalizado en marzo del 2022, siguiendo la metodología del proceso estratégico (Laluf, 2022). Estos se muestran en la Tabla 17, en donde luego de un proceso de tormenta de ideas con la línea de gerencia y directores,

determinaron los siguientes factores y su relación con los competidores, siendo nueve los más importantes.

Tabla 17

Factores críticos de éxito para el mercado de MS4M

<i>Factores</i>	<i>Peso</i>	<i>MS4M</i>	<i>MODULAR</i>	<i>WENCO</i>	<i>HEXAGON</i>	<i>CATERPILLAR</i>
1.Servicio y fidelización al cliente	11%	3.8	1.0	2.3	2.8	2.2
2.Capacidad de integración	9%	2.3	2.3	1.2	3.0	1.7
3.Investigación, desarrollo e innovación	17%	1.6	1.5	2.0	3.2	3.7
4.Conocimiento y experiencia en minería	18%	3.1	1.8	1.8	2.0	4.0
5.Capacidad y solidez financiera	8%	2.1	2.8	2.3	3.5	4.0
6.Calidad de productos certificados	9%	2.3	2.8	2.3	3.0	3.5
7.Estructura organizacional	8%	2.1	2.7	2.2	2.8	3.3
8.Presencia en el mercado global	11%	1.3	3.3	2.7	2.8	3.8
9.Liderazgo en costos	8%	1.9	2.2	2.5	2.0	1.8
Puntaje		2.3	2.2	2.8	3.2	3.2

Nota. Tomado de “Estrategic planing consultancy for MS4M”. Empresa MS4M. (2022). No publicado.

De acuerdo con esta metodología, 4 indica una fortaleza mayor, 3 una fortaleza menor, 2, una debilidad menor, y 1 una debilidad mayor. A partir de este ejercicio, MS4M determinó que sus mayores fortalezas para incursionar en el mercado conducen a un mejor servicio al cliente y a su conocimiento del negocio, desde aquí proyecta inicialmente las acciones hacia un entorno minero con las características descritas anteriormente. Sin embargo, pese a lo abundante de la información procesada, este proceso de tormenta de ideas no ha considerado lo que manifiesta (Luo & Assche, 2023) respecto a las empresas que desarrollan tecnología y el nuevo reto respecto a la realidad geopolítica en la fabricación y

comercialización de procesadores de computadores, la escases de estos componentes, tiempos de entrega con alta incertidumbre y prolongados, hace que la capacidad de MS4M de responder a su servicio al cliente se vea mermado, convirtiéndose asimismo, en una debilidad al no considerar mercados, componentes alternativos y/o alianzas con nuevos fabricantes.



Capítulo III: Metodología

En este capítulo se describirá el análisis de la implementación e impacto de las diferentes metodologías que respecto al ciclo de vida del software, y su alienación al modelo de negocio de MS4M. Considerando el desarrollo de productos y su relevancia de la operación e implicancia de priorización de funcionalidades sobre el modo de construcción de estos.

3.1. Metodología de Trabajo

Los problemas acumulados relacionados a la calidad en la ingeniería y desarrollo de software suelen denominarse deuda técnica, originada principalmente por la priorización de entregas rápidas, focalización en requerimientos funcionales y las etapas tempranas de consolidación de empresas emergentes que buscan hacerse un espacio en el mercado tecnológico (Lenarduzzi, et al., 2021), como en el caso de MS4M. Según IT Buddha (2023), la denominada deuda técnica puede estar relacionada principalmente a: a) Problemas en la calidad de código relacionados a la complejidad, duplicidad o falta de documentación, b) Problemas de arquitectura tales como incapacidad de integrarse entre componentes o módulos, falta de éstos o uso de tecnologías o versiones obsoletas, y c) Problemas en las pruebas de control de calidad, despliegues o ausencia de controles automatizados. En el caso de MS4M, que además desarrolla hardware, los problemas de la deuda técnica puede afectar a otras disciplinas como la electrónica y al ciclo de vida de los productos, retrasando la llegada a la madurez y extendiendo gastos de soporte (Vogel-Heuser & Bi, 2021).

Ante esto, IT Buddha (2023) indica que la adopción de metodologías ágiles de desarrollo incrementa la posibilidad de una efectiva gestión de la deuda técnica principalmente por su enfoque en el desarrollo iterativo, la colaboración, comunicación y la mejora continua, necesaria para el desarrollo de software actualmente. Aldaej, et al (2023)

presentó como resultado de su estudio que la adopción de prácticas de desarrollo ágil tuvo un efecto positivo en la gestión de la deuda técnica en empresas de desarrollo tecnológico, identificando las brechas desde etapas tempranas del desarrollo, mejorando la calidad del producto final, reduciendo el mantenimiento, promoviendo la colaboración y comunicación entre los miembros de los equipos de desarrollo.

En la documentación que nos proporciona (Edison, et al., 2022) nos detallan que, en las dos últimas décadas, se ha comprobado que los métodos ágiles tales como Scrum, eXtreme Programming (XP), DevOps y flow cuentan con popularidad, y en su mayoría han sido altamente efectivos entre los equipos de desarrollo de software. Sin embargo, estas metodologías ágiles sufren cambios significativos en el ciclo de vida de desarrollo cuando adoptan una integración continua y entrega rápida, el objetivo del estudio realizado por (Krey, 2022) tuvo como finalidad desarrollar una metodología para que gestione el ciclo de vida del software y DevOps para respaldar la necesidad de entregas rápidas. Bajo este estudio Scrum fue elegido como el más eficiente debido a su mejor enfoque Agile.

Debido a la falta de control sobre el flujo total de desarrollo de software es una de las principales razones por las que muchos proyectos de desarrollo de software fracasan. En este sentido Scrum ha cerrado esas brechas y su adaptabilidad ha demostrado que es el marco más sólido dentro del enfoque Agile.

Existen pronósticos que el mercado de DevSecOps se clasifica según su tipo (soluciones y servicios), modelo de implementación en la nube (pública y privada), tamaño de la organización (pequeñas, medianas y grandes empresas), verticales (telecomunicaciones, tecnología de la información, servicios de tecnología de la información, servicios financieros y bancarios), tendrán una proyección global hasta el año 2028, experimentando un crecimiento significativo, pasando de USD 10,400 millones en

2023 a USD 25,500 millones para 2028, con una tasa de crecimiento anual compuesta (CAGR) del 19,7 % durante el período de pronóstico. A raíz que las aplicaciones empresariales modernas se vuelven cada vez más complejas debido al uso de múltiples tecnologías, numerosas bases de datos y diversos dispositivos de usuario final, DevOps podría ser el único enfoque práctico para abordar eficientemente estos diversos ecosistemas. (Mehra, 2023).

Como información adicional, respecto al crecimiento de DevSecOps en el mercado global fue en 2021 con USD 3,4 mil millones y pronosticado hasta los USD 23,5 mil millones para 2028 con una tasa de crecimiento anual compuesta – CAGR (siglas en inglés) del 31,96 % durante el período de pronóstico según el reporte anteriormente citado. Para la elaboración de la propuesta de solución, se tomará como referencia la interacción de características de aspectos de desarrollo con Scrum y la entrega rápida de DevOps.

Tomando como base las practicas agiles, enfocando la automatización del proceso de desarrollo de software, beneficiando a la organización rompiendo los silos entre las áreas de desarrollo y operación, entregando software de calidad y rapidez (Almeida, et al., 2022).

Así como, la evolución de la era digital que prioriza el desarrollo de software con un enfoque a la seguridad, manteniendo la integridad, confiabilidad y disponibilidad de las aplicaciones, permitieron crear la variante segura del proceso de desarrollo de software DevOps, llamada DevSecOps, en la cual se aplica seguridad en cada una de sus etapas del proceso, mitigando las vulnerabilidades (Rajapakse, et al., 2022).

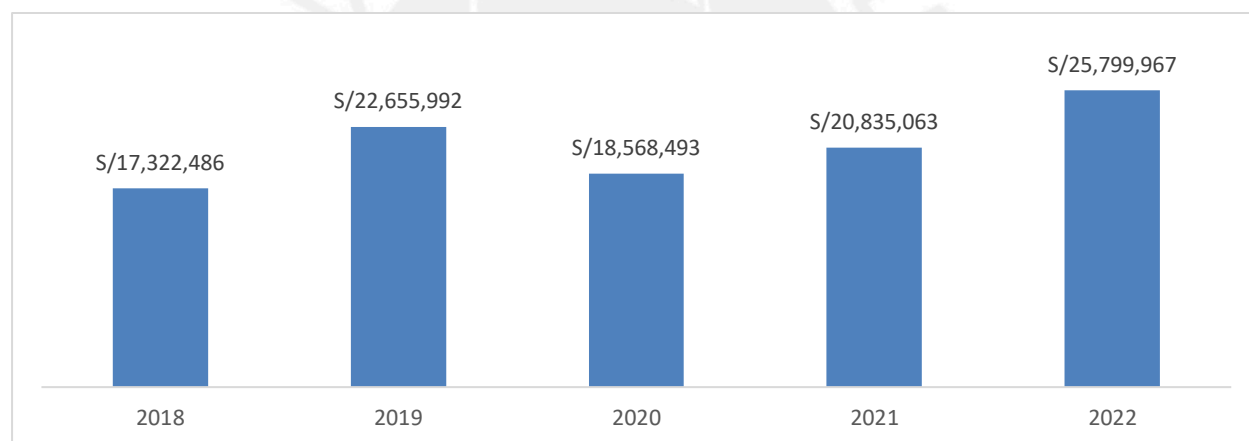
3.2. Identificación de Problemas

El crecimiento MS4M rápido alineado a su orientación hacia los clientes, implica una fuerte dependencia de capacidades del negocio relacionadas al desarrollo de nuevos

productos. A la par con los nuevos ingresos, los gastos relacionados a la planilla del personal en general también se han ido incrementado, no solamente en el área de Investigación y Desarrollo que es la más extensa, sino también en diversas áreas. La Figura 3 muestra el incremento de las ventas en una determinada proporción y de forma diferente del incremento de los gastos generales mostrados en la Figura 4, lo que determina una merma en la rentabilidad de la empresa, tal como se puede apreciar en la Figura 5.

Figura 3

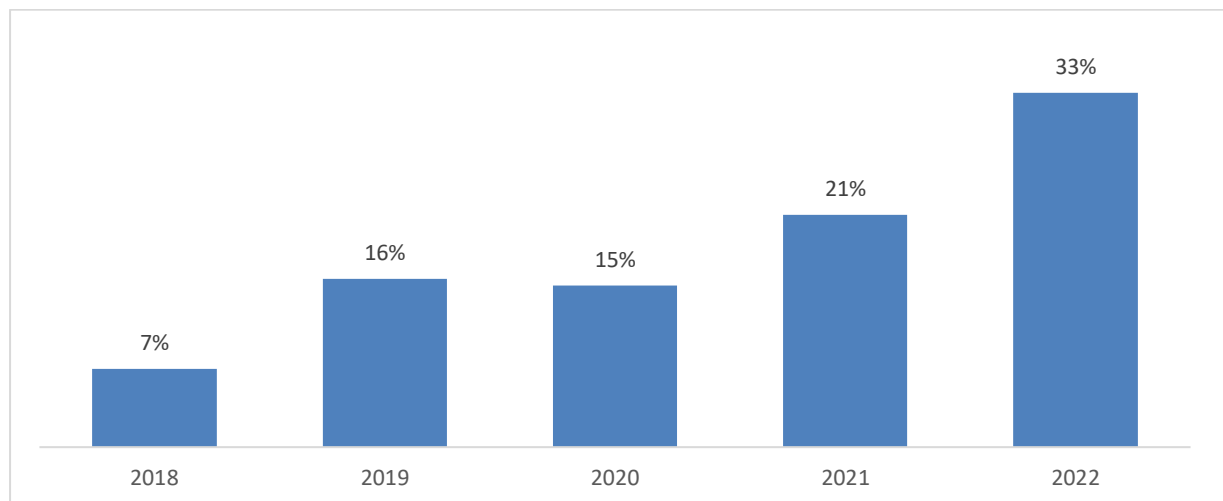
Ventas anuales en soles de MS4M 2018-2022



Nota. Extraído de Estado de resultados de MS4M de los años del 2018 al 2022. MS4M. No publicado

Figura 4

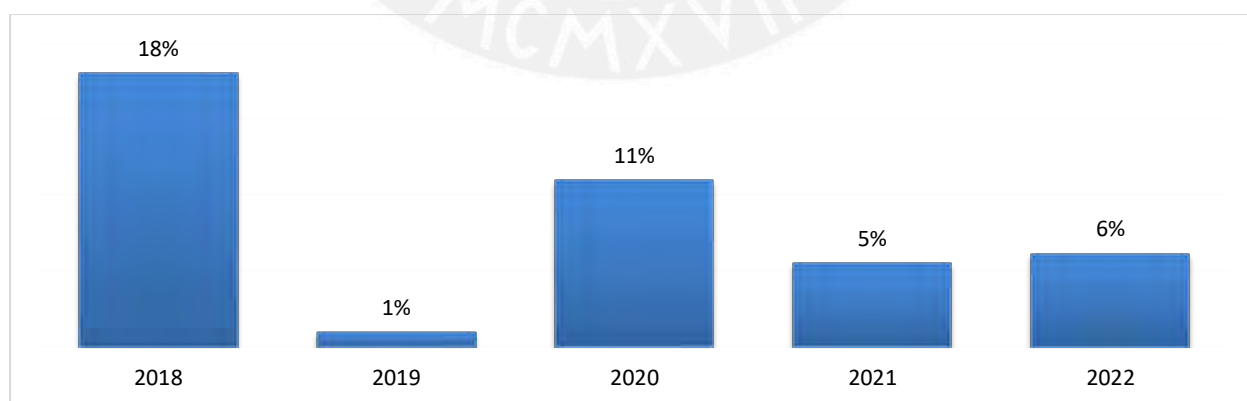
Evolución de gasto de personal respecto a ingresos de MS4M 2018-2022



Nota. Extraído de Estado de resultados de MS4M de los años del 2018 al 2022. MS4M. No publicado

Figura 5

Evolución de la utilidad neta respecto a ingresos de MS4M 2018-2022



Nota. Extraído de Estado de resultados de MS4M de los años del 2018 al 2022. MS4M. No publicado

En la tabla 18, se detalla el estado de resultados del año 2022 de MS4M, evidenciando el nivel de gasto y descomposición en gastos administrativo, así como en investigación y desarrollo. A partir de la tabla, se concluye que la política de MS4M orientada a la satisfacción y respuesta a las demandas de sus clientes exige invertir significativamente en investigación y desarrollo. Se estima que dicho gasto en el personal de investigación y desarrollo equivale el 14% del total de los ingresos del 2022.

Tabla 18

Estado de resultados año 2022 de MS4M en dólares americanos

<i>Estado de Resultados 2022</i>	
Ingresos por servicios	4,087,687
Ingresos por venta de productos	2,639,018
Otros ingresos	14,000
Descuentos por pagos anticipados	-8,218
Total de ingresos	6,732,487
Costos de servicios	-1,536,855
Costos de producción	-1,121,355
Total de costo de ventas	-2,658,209
Margen bruto	4,074,278
% de Margen bruto	60.52%
Gastos de I&D	-934,712
Gastos administrativos	-1,254,046
Gastos de venta	-1,137,286
Total de gastos	-3,326,044
Margen operativo	748,233
% de Margen operativo	11.11%
Ingresos financieros	15,228
Costos financieros	-58,350
Diferencia de cambio neto	-125,706
Ingresos antes de impuestos	579,406
Participaciones	-57,941
Impuestos	-170,925
Ingresos netos	350,540
% de Ingresos netos	5.21%

Nota. Tomado de *Informe General para Directorio 2022 MS4M*. Empresa MS4M. (2022),

(No publicado)

A lo largo de la historia de MS4M, se destaca el desarrollo de productos enfocados a sus clientes, es decir, consideran como prioridad importante la funcionalidad y aplicabilidad

de sus productos en cada operación implementada. Actualmente la empresa, cuenta con una estructura de desarrollo de productos y servicios que generan altos gastos operativos tras su implementación en sus clientes, en el listado de incidentes de soporte de sistemas escalados de los años 2022 y 2023 (tabla 6 y tabla 7), se determina que el nivel de escalamiento promedio ascendía al 20% de las peticiones de soporte, lo cual involucra al personal de desarrollo para su resolución. Esto conlleva a que la calidad de los productos y servicios de MS4M se vea disminuida, y que los productos se perciban ante los clientes como inestables y afecten la competitividad de la empresa, de igual manera, el mantenimiento y soluciones para los incidentes en cada operación genera un sobrecosto notable, lo que perjudica en el cumplimiento de los objetivos estratégicos.

Entre las perspectivas consideradas en los objetivos estratégicos de MS4M (tabla 16), se consideran, por ejemplo: La financiera, MS4M actualmente posee un EBIDTA del 15% y tiene como objetivo superar el 40% mismo para el 2027; en la perspectiva de Procesos, la empresa prevé colocar una operación minera en cada continente, con soluciones autónomas y/o a través del uso la Inteligencia Artificial. De acuerdo con una encuesta realizada por McKinsey en 2020 a cincuenta CIOs de compañías de servicios financieros y tecnológicos, obtuvo que el 30 por ciento consideran que más del 20 por ciento de su presupuesto técnico dedicado al desarrollo de nuevos productos se desvía en la resolución de problemas relacionados con la deuda tecnológica, lo cual impacta significativamente en el crecimiento de las organizaciones.

En el caso de MS4M, el 17% del tiempo del personal de desarrollo de nuevos productos es invertido en corregir problemas escalados del nivel 1 de atención, lo que impacta en la generación de nuevos productos y en un mayor número de personal involucrado, que cubra la brecha que demanda el soporte. En este sentido, la organización

considera que el cumplimiento de los objetivos estratégicos requerirá que se ejecuten acciones que involucren directamente al origen de los problemas, los cuales se relacionan a procesos internos de desarrollo y la aplicación de buenas prácticas en todo el ciclo de desarrollo de productos, el listado de los problemas generales identificados por la organización se muestra en la tabla 13 y 10.

3.3. Determinación de Problemas principales

MS4M, como parte de un análisis interno, realizó una valoración de problemas internos y externos (tabla 13 y 10), en la cual se destaca la existencia de una relación entre los problemas y el proceso de desarrollo de software. En la relación de amenazas identificadas se considera que el 30% de estas puede ser mitigadas a través de la aplicación de buenas prácticas y metodologías de desarrollo, tales como:

Tabla 19

Amenazas relacionadas al proceso de desarrollo de software

<i>Amenazas</i>
1. Existencia de restricciones-barreras por ser de origen peruano/ latinos
2. Fuga de conocimientos y/o talentos
3. Existe alta demanda de productos y servicios certificados y con propiedad intelectual a nivel internacional.

Nota. Tomado de “*Estrategic planing consultancy for MS4M*”. Empresa MS4M. (2022), (No publicado)

Las amenazas identificadas en el ejercicio relacionadas al origen latino de los productos, tiene un fuerte componente concerniente a la calidad, que con un continuo escalamiento de soporte al nivel 3 de atención impedirá la evolución de los productos y, una recurrencia de problemas en los productos terminará por afectar dicha percepción y reforzarla. De la misma forma, la

alta demanda de correcciones generará un clima de frustración y desorden que incrementa el riesgo de renuncias y la capacidad de retener personal clave se vea disminuida. Por lo mismo, la demanda de productos tecnológicos certificados en cuanto a calidad y seguridad de la información relegará los productos de MS4M en el mediano plazo si no emprende procesos que consoliden la madurez de sus productos; de igual manera, en la relación de debilidades de la empresa, se identificó que el 50% conlleva relación al proceso de desarrollo de productos, encontrándose las siguientes:

Tabla 20

Debilidades relacionadas al proceso de desarrollo de software

<i>Debilidades</i>	
1.	Procesos y procedimientos internos no están definidos (No hay ERP, en logística, etapas de proyectos)
2.	Falta de la gestión del conocimiento (una parte es la documentación técnica) y comunicación (los que hay no están socializados)
3.	Débil gestión de QA/QC
4.	Poca capacidad de retención de talentos y falta de gestión de personas
5.	Productos en etapa no de madurez (“evolución”) que ocasiona una alta demanda de soporte.

Nota. Tomado de “*Estrategic planing consultancy for MS4M*”. Empresa MS4M. (2022), (No publicado)

MS4M realizó una valoración del estado de madurez de sus procesos, que implican la consolidación de prácticas de las principales capacidades de negocio necesarias para la industria según el marco de referencia APQC *Process Classification Framework PCF* (2022) obteniendo los resultados mostrados en la tabla 21.

Tabla 21*Nivel de madurez de los procesos en MS4M*

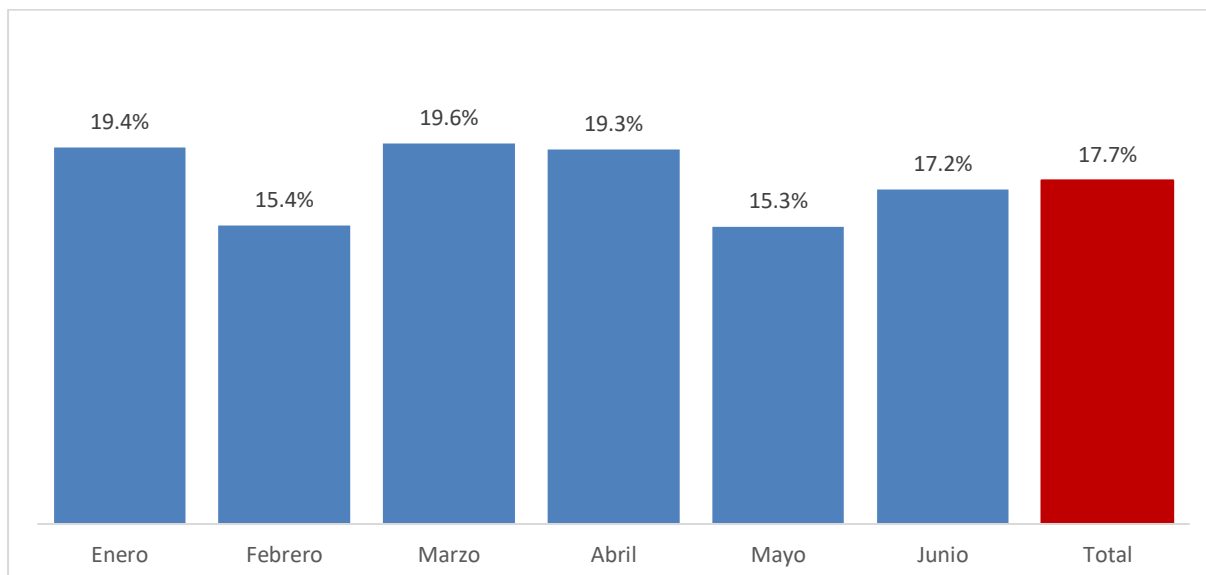
<i>Nivel de madurez</i>	<i>Definición</i>	<i>Detalle</i>	<i>% de Procesos de MS4M</i>
0	Inexistente	No se aplica la administración de procesos	15%
1	Inicial	Proceso se cumple mediante actividades incompletas	40%
2	Repetible	Proceso se cumple mediante actividades básicas pero completas	40%
3	Definido	Proceso se cumple de manera organizada y es documentado	5%
4	Gestionado	Proceso está bien definido y se mide su desempeño	0%
5	Optimizado	Proceso se cumple con mejora continua	0%

Nota. Tomado de “*Estrategic planing consultancy for MS4M*”. Empresa MS4M. (2022), (No publicado)

Esta evaluación muestra la brecha que existe en MS4M en cuanto a la consolidación de la formalidad de sus procesos en general, incluyendo los pertenecientes a su cadena de valor y, por ende, al desarrollo y producción de sus productos tecnológicos. MS4M contabiliza los activos intangibles generados a partir del desarrollo de productos tecnológicos, consistentes en el capital invertido expresados en horas hombre de los desarrolladores, insumos y equipos para prototipado. En este sentido, MS4M mantiene el registro del tiempo en el que el personal se encuentra en etapas de investigación, soporte de nivel 3 y en etapas de desarrollo; siendo la última categoría la legítimamente considerada para formar parte del activo, las primeras se contabilizan como gasto en el estado de resultados financieros. La evaluación correspondiente a junio del año 2023 se muestra en la Figura 6, donde se puede determinar que, en el presente año, el personal del área de Investigación y Desarrollo invirtió el 17.7% del tiempo total en brindar soporte de nivel 3 a los productos de MS4M.

Figura 6

Horas de trabajo (%) del personal invertidas en soporte por mes (2023)

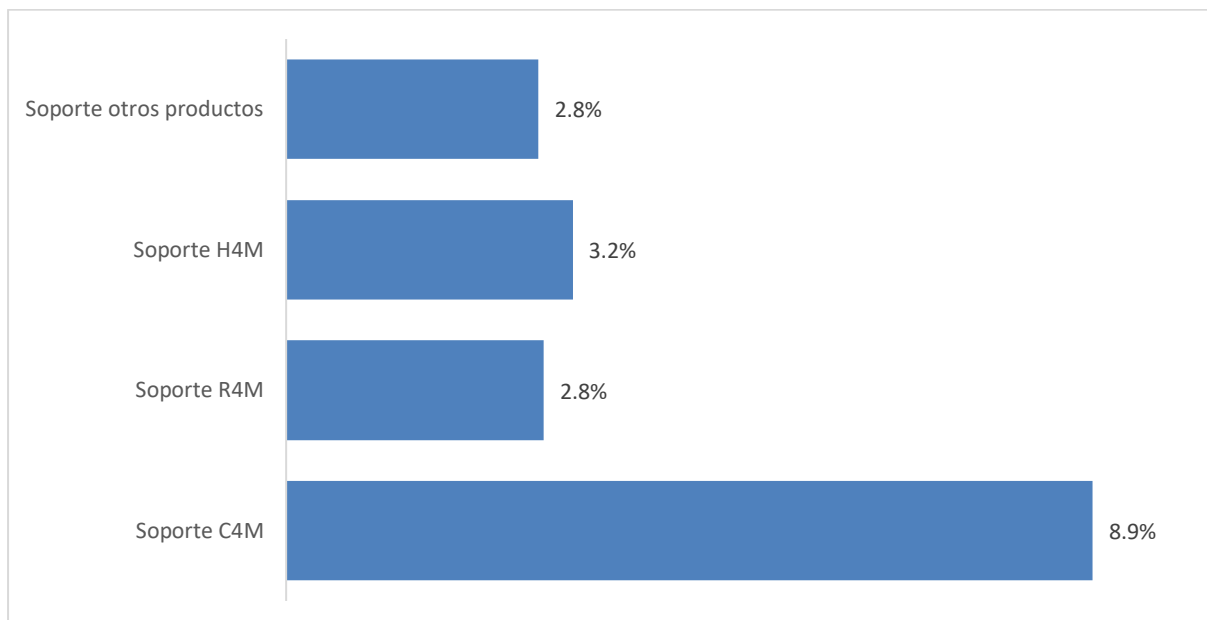


Nota. Extraído de informes de registro mensual de horas de trabajo de personal de I&D en MS4M 2023. MS4M. No publicado

Significa que, siendo la planilla a junio del 2023 del área de 45 ingenieros, 8 se dedican de forma exclusiva a brindar soporte de nivel 3 y no a desarrollar productos, formando un gasto considerable que se desprende del activo intangible. Los productos que han demandado mayor cantidad de soporte se encuentran descritos en la Figura 6, en donde se muestra que el producto principal de MS4M denominado Control 4Miners – C4M, registra una considerable cantidad de horas superior al resto de productos desplegados.

Figura 7

Horas de trabajo (%) del personal invertidas en soporte 2023, por producto



Nota. Extraído de informes de registro mensual de horas de trabajo de personal de I&D en MS4M 2023. MS4M. No publicado

El problema central que viene presentando la empresa MS4M se debe a la falta de procesos definidos, la alta demanda de soporte, acumulación de deuda técnica y costos elevados en soporte y mantenimiento, se encuentran obstaculizando su crecimiento, rentabilidad y competitividad, esta situación se caracteriza por los siguientes elementos:

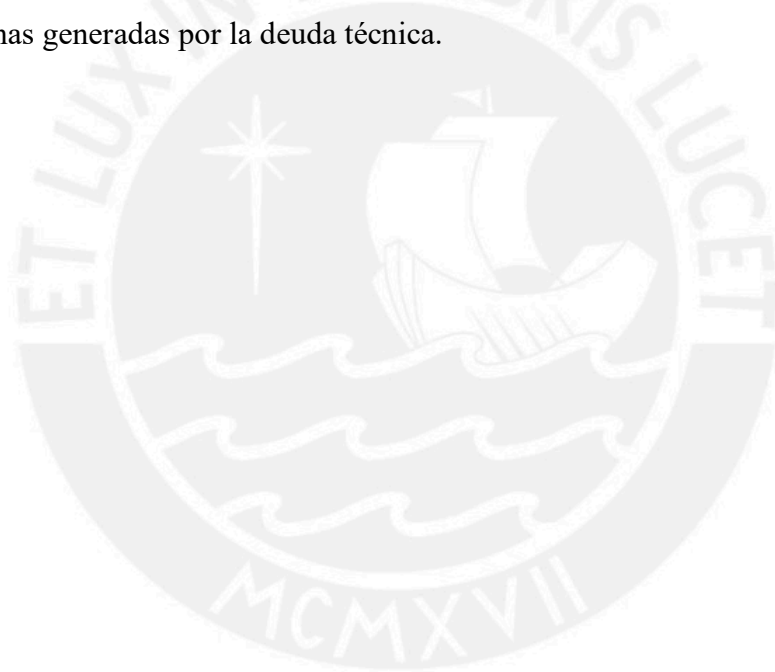
Demanda reactiva: El tiempo y los recursos del equipo de desarrollo se ven consumidos por la resolución de problemas en productos existentes, limitando su capacidad para crear nuevos productos y mejorar los existentes.

Procesos ineficientes: La falta de definición y madurez en los procesos de desarrollo genera inconsistencias, retrasa la entrega de productos y aumenta el riesgo de errores.

Acumulación de deuda técnica: La constante necesidad de corregir problemas y la falta de inversión en la mejora de los procesos conducen a una acumulación de deuda técnica, lo que dificulta el mantenimiento y la evolución de los productos.

Costos crecientes en soporte y mantenimiento: La combinación de la alta demanda de soporte, la ineficiencia en los procesos y deuda técnica genera un aumento sostenido de los costos, afectando la rentabilidad de la empresa.

Tras esta identificación MS4M, determinará las acciones y actividades necesarias para orientar los esfuerzos en fortalecer y potenciar el proceso de desarrollo de software, buscando reducir las brechas generadas por la deuda técnica.



Capítulo IV: Revisión de la literatura

En este capítulo se analizan las metodologías de desarrollo que pueden ser aplicadas a la construcción de productos de MS4M, considerando los marcos de referencia en gestión de proyectos de ingeniería de software, su implementación e impacto en el ciclo de vida del software, así como principios de desarrollo, alienado al modelo de negocio de MS4M y la construcción de productos. De esta manera se describirán los diferentes marcos de referencias, los principios del desarrollo de software aplicables y metodologías que aporten en dicho ciclo.

4.1. Estado del Arte

Desde los inicios del desarrollo de software (alrededor de 1960), la comunidad de ingeniería de software (Grupo de investigadores, académicos, profesionales y otros interesados en el desarrollo y mejora de los procesos de desarrollo de software), consideró la escalabilidad y productividad de las aplicaciones, enfocando su investigación en la reutilización y resolución de errores en estas, promoviendo técnicas y/o recursos que aporten la práctica del desarrollo de software. En este sentido el progreso de las herramientas utilizadas y las nuevas técnicas de desarrollo, permiten crear software cada vez más complejo conforme a la evolución y demanda del mundo actual (Gerostathopoulos, et al., 2021).

Este desarrollo requiere considerar el ciclo de vida del software, la cual está conformado generalmente por cuatro fases comunes: (1) Fase de Requisitos: en la cual se identifican y gestionan los requisitos que deberá cumplir la aplicación. (2) Fase de Diseño: Donde el software es esquematizado y diseñado a nivel visual y de código. (3) Fase de Código: En el cual desarrollo inicia y se implementa el software. (4) Fase de Pruebas: Donde se verifica que la operatividad de la aplicación, y las funcionalidades solicitadas se encuentren implementadas. (Dlamini, et al., 2022)

En este sentido, es importante que la ejecución de las actividades que se definan en cada fase del proceso de desarrollo de software, debe considerar el enfoque basado en el valor (ingeniería de Software basada en Valor – VBSE por sus siglas en inglés), el cual considera asegurar que estas actividades, consideren a lo largo de todo el ciclo de vida de software el cumplimiento de las necesidades de los interesados, usuarios finales y clientes, los cuales conllevan a realizar seguimientos a aspectos técnicos, funcionales, económicos, sociales y éticos, buscando maximizar del valor entregado por la aplicación a lo largo de las fases o etapas del ciclo de vida del software. (Rodríguez, et al., 2022)

4.1.1. Sobre los marcos de referencia para el desarrollo de software

Los marcos de referencia o estándares son herramientas utilizadas en la industria del desarrollo de software para mejorar la eficiencia de los procesos involucrados en la creación y mantenimiento de software. Estos marcos proporcionan pautas y mejores prácticas que ayudan a las empresas a estandarizar sus procesos y garantizar la calidad de sus productos. Al seguir estos marcos, las empresas pueden optimizar sus actividades de desarrollo, reducir los errores y aumentar la productividad. Estos estándares abarcan diferentes áreas, como la gestión de proyectos, la ingeniería de software y la calidad del producto. Sin embargo, se ha observado que las pequeñas empresas de software enfrentan dificultades para adoptar estos marcos, lo que puede limitar su capacidad para mejorar la eficiencia y la calidad de sus productos (Tuape, et al., 2021).

En este sentido al considerar la adopción de los marcos de referencia o estándares en el contexto de la implementación del diseño centrado en el usuario en el desarrollo de software ágil. En primer lugar, se recomienda seleccionar un marco de referencia que se adapte a las necesidades específicas del proyecto y del equipo de desarrollo. Además, es

importante involucrar a los usuarios finales en el proceso de diseño y desarrollo para garantizar que se satisfagan sus necesidades y expectativas. También se sugiere establecer un proceso iterativo y colaborativo que permita la retroalimentación constante y la mejora continua del producto. Por último, se destaca la importancia de la formación y capacitación del equipo de desarrollo en el marco de referencia seleccionado para garantizar su correcta implementación y uso. (Dudysheva & Skopin, 2023).

4.1.2. Principios del desarrollo de software

Existen muchos principios para el desarrollo de software en la actualidad que usan como marcos de referencia metodologías ágiles y otros tradicionales usados por desarrolladores para la creación de software. Estos principios han permitido fortalecer el desarrollo de software (Damyanov, et al., 2024).

Uno de los primeros principios está relacionado a la modularidad y funcionalidad del software a desarrollar, pues la modularidad es definida como la mínima unidad independiente a ser interconectada con otros módulos, para que en conjunto, satisfaga las necesidades y requerimientos del sistema (Damyanov, et al., 2024), y cuya relación deriva en otro llamado *Principio de Responsabilidad Única*, introducido por Robert C. Martin atribuyendo un único propósito y razón de cambio a cada módulo desarrollado en el sistema (Sánchez, 2023).

Asimismo, Robert C. Martin quien introduce otro principio importante y popular en el desarrollo de software actual, como lo es el de cerrado-abierto (OCP) que, establece que los módulos deben estar lo suficientemente cerrados para no permitir un cambio en la codificación, pero a su vez, lo suficientemente abiertos para permitir extensiones en base a nuevos requerimientos, es decir, que la reutilización de los algoritmos llevados a código sea no intrusiva (Swan, et al., 2022).

La abstracción es otro principio dentro del marco tradicional de desarrollo de software e introducido a inicios del presente siglo, que establece que, durante la ingeniería de software. Por esta razón es importante priorizar los detalles que conllevan a la naturaleza del código y su razón, dejando las características irrelevantes para el nivel en el que se está trabajando (Albonico, et al., 2023).

Relacionado a la modularidad se derivan otros principios como la del Principio de Sustitución de Liskov (LSP) que establece que las clases mayores pueden ser reemplazadas o representadas por cualquier elemento que la conforma o sub clase (Aratchige, et al.,2024), correspondiendo al desarrollo de software orientado a objetos. Ellos también definen otro Principio de Segregación de Interfaces (ISP), que establece que deben considerar como objetivo la satisfacción al usuario final, así como evitar despliegues de gran volumen, que no aporten utilidad a los stakeholders, de esta manera se considerará las preferencias de estos en funcionalidades básicas con importante impacto (Robillard, 2022).

Otro principio importante derivado de la modularidad es el Principio de Inversión de Dependencia (DIP), que implica que los módulos son interdependientes entre sí. Este principio aporta flexibilidad y escalamiento para labores de soporte y evoluciones rápidas de todo el sistema (Holdener, 2022).

Existen otros principios, que de acuerdo al contexto de desarrollo pueden ser aplicados. No obstante, todos estos otorgan guías que aportan a la calidad en el diseño y la estructura de las aplicaciones, así también como: la gestión del conocimiento, la flexibilidad, mantenimiento y control de calidad durante o después del desarrollo, previniendo costos innecesarios en intervenciones posteriores a la entrega del producto final (Afshar, 2022).

4.1.3. Metodologías del desarrollo de software

El desarrollo de software a lo largo de un proyecto puede considerar diversas metodologías, las aplicables dependerán de las naturales de este, así como de su objetivo. Estas metodologías proveen una sistemática secuencia de eventos o etapas (pueden o no ser lineales) para obtener un producto o aplicación. La metodología representa el “cómo” e identifica como se desarrolla cada actividad en cada etapa, cómo se presentan las actividades y productos, y finalmente cómo se generan, es decir, deben de seguir un denominado “ciclo de vida” (Laplante & Kassab, 2022).

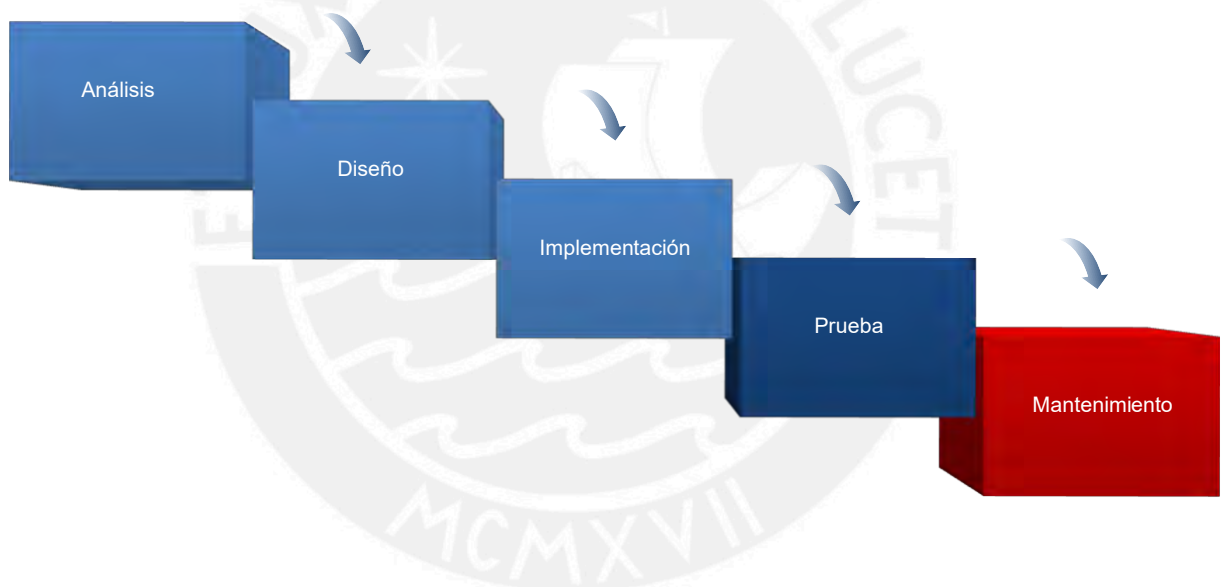
Las principales metodologías utilizadas en la actualidad se agrupan en enfoques tradicionales, que son las más longevas, y las denominadas ágiles. Hay muchos más enfoques tradicionales que ágiles en la actualidad, tales como el de cascada, interactivo, incremental, en espiral, evolutivo, entre otros. Entre todas las metodologías tradicionales son las más populares las de cascada, y tiene características muy marcadas, así como dependencias para su correcto uso (Edison, et al., 2022).

La metodología en cascada plantea un ciclo de vida lineal, con etapas dependientes del inmediato anterior, cuyo progreso depende de la etapa anterior diseñada, construida y puesta a prueba. La secuencia similar al de una cascada de agua real, comprende etapas consecutivas de análisis del problema, diseño, implementación, puesta a prueba y mantenimiento (Aroral, 2021). Sin embargo, tiene muchas restricciones a razón de dependencia de requerimientos funcionales muy bien definidos, escritos y planificados, así como de una casi nula variabilidad durante todo el ciclo de vida (Dursun, et al., 2021). Un eventual cambio en las restricciones o requerimientos lo hace costoso e improductivo, es altamente sensible a los cambios, pero es muy útil cuando se trata de proyectos de gran tamaño y gran complejidad, con tiempos y presupuestos limitados y previamente definidos.

Presenta muchas ventajas adicionales tales como la ausencia de experiencia previa o de capacitaciones adicionales, así como desventajas como la premura de información previa al inicio y de la falta de retroalimentación al usuario final sobre la utilidad del producto en construcción, que crea una dependencia de la terminación del producto para realizar posibles cambios, lo que implica un reinicio del ciclo de vida. En la Figura 8 se muestra la secuencia del enfoque tradicional de cascada.

Figura 8

Secuencia de la metodología tradicional de cascada



Nota. Tomado de *Waterfall Process Operations in the Fast-paced World: Project Management Exploratory Analysis*. Aroral (2021).

Sin embargo, en un entorno cambiante como en la actualidad, la metodología tradicional no es aplicable de manera íntegral, Aroral (2021) indica que la capacidad de predecir el comportamiento del entorno es cada vez más pequeño y por lo tanto, los métodos tradicionales dieron paso a metodologías que adoptaron el nombre de ágiles precisamente por

su capacidad de adaptarse a este nuevo escenario y representar el más popular método para desarrollar software en la actualidad.

La metodología ágil representa un enfoque ligero, para superar las limitaciones de dependencia secuencial de la metodología tradicional y adaptarse a cualquier cambio en los requerimientos previamente definidos, y lo realiza mediante la gestión de tareas, subdividiendo, coordinando y ejecutando a través de un cierto número de valores y principios. En esta metodología, el ciclo de vida deja de ser lineal y se vuelve cíclico (Mishra & Alzoubi, 2023).

Petit & Marnewick (2021) han resumido los principales elementos clave de la metodología ágil:

- Los usuarios finales están involucrados permanentemente en el ciclo de vida
- El producto se desarrolla gradualmente, con pequeños entregables que le aportan valor al usuario final
- Cada miembro del equipo de desarrollo rinde cuentas de su desempeño en pequeños intervalos de tiempo y ciclos de revisión.
- Fuerte enfoque en la documentación del trabajo realizado.
- El usuario final define el producto final, acorde a su medida

Al igual que las metodologías tradicionales, las denominadas ágiles mantienen restricciones que son de carácter interno, en forma de código, o “manifiesto” que congrega valores y principios de colaboración y comunicación entre los miembros de los equipos. El denominado manifiesto ágil comprende 4 valores y 20 principios con el objetivo de generar valor al usuario final en sus interacciones mediante una estrecha comunicación y colaboración, lo que le permite manejar la incertidumbre asociada a los cambios del entorno

y reducir la complejidad (Zielske & Held, 2021). La tabla 29 resume las principales diferencias entre las metodologías tradicionales y las ágiles.

Tabla 22

Enfoque tradicional versus enfoque ágil

<i>Parámetro</i>	<i>Enfoque tradicional (Waterfall)</i>	<i>Enfoque ágil</i>
Facilidad de modificación	Dura	Fácil
Enfoque en el desarrollo	Predictivo	Adaptativo
Orientación del desarrollo	Orientado al proceso	Orientado al cliente
Tamaño del proyecto	Grande	Pequeño o mediano
Escala de planificación	Largo tiempo	Corto tiempo
Estilo de gestión	Control y mando	Liderazgo y colaboración
Aprendizaje	Aprendizaje continuo durante el desarrollo	Aprendizaje es secundario al desarrollo
Documentación	Alta	Baja
Presupuesto	Alto	Bajo
Involucramiento del cliente	Bajo	Alto
Entrega del producto	Al final del proyecto	Iterativa/incremental
Respuesta al cambio	Difícil	Fácil
Nivel de riesgo	Alto	Bajo
Comunicación	Formal	Informal
Organización del equipo	Jerárquica	Autoorganizada

Nota, Adaptado de Structured software development versus agile software development: a comparative analysis. Mishra, A., & Alzoubi, Y. I. (2023). International Journal of System Assurance Engineering and Management

4.1.4. Métodos de desarrollo de software

Los métodos son enfoques amplios que designan una estructura determinada de organización y dirige el proceso de desarrollo de software a lo largo del ciclo de vida (Gutiérrez-Ríos, et al.,2021). Dentro de las metodologías tradicionales se pudo observar las

características predominantes del método de cascada como principal representante de dicho grupo, sin embargo, en cuanto a las metodologías ágiles son muy populares diversos métodos con diferentes características.

El enfoque ágil es ligero, que se adapta a cambios, con objetivos específicos de centrarse en el cliente y reducción de costos por retrabajos, basado en comunicación y coordinación exhaustiva para la gestión granular de tareas que agregan valor incremental, con principios y valores guía a través de todo el proceso. Los métodos ágiles son una combinación, cada una diferente en denominación y en forma, de prácticas que definen detalladamente las tareas de los desarrolladores en el día a día, y que en su conjunto apoyan la filosofía de la metodología ágil expresada en los principios y valores (Mishra & Alzoubi, 2023).

Zielske & Held (2021) consideran que los métodos ágiles más ampliamente utilizados son Scrum, Kanban, XP, FDD y Crystal. Scrum es una combinación de adaptación y manejo de cambios junto con entregas parciales que agregan valor, todo a través de un enfoque de interacciones de equipo a través de ciclos pequeños (o “sprints”) y reuniones de sincronización de diferentes jerarquías y frecuencias, entre los miembros de un equipo liderado por un “dueño de producto” y de un “maestro Scrum” (Tashtoush, et al., 2021).

El método Kanban se nutre de los principios de *Lean Manufacturing* para enfatizar la eliminación de desperdicios, enfocarse en el aprendizaje y la holgura para la toma de decisiones. Se concentra en la visualización de las tareas clasificadas secuencialmente en un tablero, donde cada secuencia expresa en columnas el progreso de cada tarea usando las tarjetas Kanban, lo que permite un adecuado flujo de trabajo (Gaete, et al., 2021).

El método XP o *Extreme Programming* está basado en la simpleza, comunicación, osadía, retroalimentación y respeto (Shastri, et al., 2021). Se pone mucho énfasis en los ciclos

en todos los niveles, lo que implica sincronizaciones con frecuencias regulares no mayores a una hora diariamente para compilaciones regulares. Usa artefactos sofisticados como la refactorización, la integración continua y el desarrollo basado en pruebas (Zielske & Held, 2021).

De manera similar a XP, existe el método FDD (Feature Driven Development) basado en las características del proyecto, con un enfoque interactivo e incremental. Se divide en 5 fases que se inician con la identificación de las características principales para una consecuente subdivisión en características más pequeñas para asignación de tareas, luego se ejercen diseños de arquitectura y planes para las menores características que determina la última etapa de entregas y pruebas en iteraciones cortas conforme se vayan alcanzando e incorporando otras características. Para ello es importante la sincronización y entrega continua de paquetes tangibles de producto con incrementos graduales para completar todas las características previamente definidas (Zielske & Held, 2021).

El método Crystal es un conjunto de diferentes sub métodos, niveles de tamaño o complejidad debido a una criticidad previamente definida, en base a la satisfacción del cliente y rentabilidad del proyecto (Schramm, et al., 2023). Todos los sub métodos (Crystal Clear, Crystal Yellow, Crystal Orange) comparten las mismas reglas y valores tales como el enfoque en el aprendizaje, comunicación y colaboración, la entrega frecuente, involucramiento del usuario final y reuniones de sincronización frecuentes y reflexivas (Zielske & Held, 2021).

Los métodos ágiles descritos fueron diseñados e implementados para trabajos considerando sólo un equipo por proyecto, sin embargo, existen otros métodos que integran el trabajo de diferentes equipos ágiles involucrados en un solo producto, donde son necesarias niveles de escalamiento. Son reconocidos en este entorno: SAFe (Scaled Agile Framework) y

LeSS (Large Scaled Scrum), basados en Scrum. En SAFe, el objetivo es la gestión y apoyo en equipos escalables de más de cien desarrolladores con enfoques Lean y ágil, con niveles de organización basados en criterios de gobernanza y de gestión de productos, usando herramientas como el Agile Release Train para la integración y de gestión de conocimiento como el Inspect and Adapt (Menzel, 2021). Para menos de 8 equipos LeSS ofrece integración basados en reglas y guías de escalamiento sin herramientas adicionales más que Scrum (Páramo Sepúlveda, 2023).

Las tendencias de los métodos ágiles a gran escala y el desarrollo de infraestructura en la nube, fuerzan a los métodos ágiles a incluir el aspecto operacional dentro del ciclo de vida del desarrollo del software, de esta manera, nace en Europa DevOps como una alternativa de integrar a los receptores de los entregables dentro del proceso de creación del producto final, donde la integración continua y las pruebas no podían amortiguar las exigencias de las operaciones convirtiéndose en cuellos de botella que paralizaban el desarrollo (Karunarathne, et al., 2024). De la misma forma, DevSecOps incorpora el elemento de seguridad de la información en el desarrollo con el método DevOps, para asegurar la confidencialidad, integridad y disponibilidad del producto final desde una temprana etapa del ciclo, (Akbar, et al., 2022).

A lo largo del proceso de desarrollo de software es importante considerar la implementación y/o aplicación de diferentes prácticas que involucran devops, las cuales implican: implica: (1) la adopción de un modelo DevSecOps, buscando integrar la seguridad en cada etapa de desarrollo de aplicaciones, (2) las políticas de gobernanza y políticas de seguridad deben propiciar el entorno adecuado para devops, (3) el manejo de incidencias de vulnerabilidades en las aplicaciones desarrolladas, deben ser gestionadas de manera oportuna, (4) es importante que el proceso de automatización de identificación vulnerabilidades tenga

mecanismos o herramientas que permitan proteger el entorno de desarrollo. (5) asimismo, la revisión y testeado de pruebas en los equipos de seguridad físicos o externos que pueden afectar el desarrollo de las aplicaciones (Galvan-Cruz, et al., 2024).

De acuerdo con Desai & Nisha (2021), para la inclusión de la seguridad en el ciclo de vida de desarrollo de software y su entorno (DevSecOps), involucra que los procesos de integración, entrega y despliegue continuo sean automatizados y que su monitoreo de seguridad integrada y constante permite garantizar aspectos de seguridad en el desarrollo de software, mismo que debe ser soportado por las herramientas tecnológicas que aporten cada uno de estos procesos, entre ellos pueden considerarse el uso de contenedores, microservicios, infraestructura como código.

La implementación de estas herramientas, tecnologías, actualización de procesos y formas de trabajo, conllevan a asumir retos y desafíos que deberán ser asumidos por las organizaciones, las cuales deberán integrar en su gestión tecnológica, la seguridad de la infraestructura, la seguridad en cada etapa de desarrollo, despliegue, así como la gestión de las amenazas y protección del software. De esta manera se busca integrar la seguridad en todo el ciclo de vida de desarrollo de software, abordando desde los inicios de este proceso las vulnerabilidades potenciales, así como fortalecer la colaboración entre equipos de seguridad y desarrollo para garantizar la detección de incidencias (Anjaria & Kulkarni, 2022).

Todos estos métodos ágiles, al ser aplicados en cualquier organización en forma exclusiva, combinada o hasta híbrida, deben de ser puestos a prueba ante determinados criterios en base a su idoneidad. Es decir, deben de afrontar una realidad empresarial existente con estándares y procesos contrarios o favorables a la metodología ágil, maduros o inmaduros, y un nivel cultural favorable o no, en una primera instancia. Posteriormente los proyectos deben ser particionados a nivel granular tomando criterios de separación en base al

riesgo, complejidad y la interdependencia, y evaluar los niveles de incertidumbre sobre el alcance para determinar si la metodología ágil o la tradicional puede ser aplicada. En suma, muchos enfoques determinan que las grandes organizaciones adoptan metodologías híbridas en base a su realidad, y la del proyecto, con un fuerte componente ágil en dicha combinación. (Edison, et al., 2022).

4.1.5. Herramientas utilizadas

Las metodologías ágiles de desarrollo de software, como Scrum, Kanban y Extreme Programming (XP), se centran en la entrega continua de software de alta calidad a través de la colaboración y la adaptabilidad. Para lograr estos objetivos, se utilizan prácticas específicas como la integración continua, la entrega continua y la automatización de pruebas. Para implementar estas prácticas, se utilizan herramientas como Git, Subversion y Mercurial para el control de versiones de código; Ant, Maven y Gradle para la construcción; Puppet, Chef, Salt, Ansible y CFEngine para la automatización, Jenkins, Bamboo, Cruise Control, Hudson, Sysphus, Hydra y TeamCity para la integración continua, y Nagios, Zabbix, Sensu e Icinga para el monitoreo del rendimiento (Yang, et al., 2023).

Estas herramientas posibilitan la implementación de metodologías y prácticas ágiles en el desarrollo de software. Las herramientas de control de versiones permiten a los equipos de desarrollo colaborar en el mismo código fuente y mantener un historial de cambios. Las herramientas de construcción automatizan el proceso de construcción y compilación del software, lo que permite a los equipos de desarrollo compilar y construir el software de manera más eficiente y confiable (Melgar, et al., 2021). Las herramientas de automatización permiten a los equipos de desarrollo automatizar la configuración y el aprovisionamiento de servidores y aplicaciones, lo que acelera el proceso de entrega de software. Los servidores de integración continua automatizan el proceso de integración y entrega continua, lo que permite

a los equipos de desarrollo integrar y entregar el software de manera más rápida y confiable. Las herramientas de monitoreo permiten a los equipos de desarrollo detectar y solucionar problemas de manera proactiva, lo que mejora el rendimiento y la disponibilidad del software en producción.

En DevSecOps, se utilizan diversas herramientas para integrar la seguridad en el ciclo de vida del desarrollo de software. Algunas de estas herramientas incluyen: herramientas de análisis estático de código, como SonarQube y Checkmarx, que permiten identificar vulnerabilidades en el código fuente; herramientas de análisis dinámico de seguridad, como OWASP ZAP y Burp Suite, que permiten identificar vulnerabilidades en la aplicación en tiempo de ejecución; herramientas de gestión de vulnerabilidades, como Jira y Trello, que permiten gestionar y priorizar las vulnerabilidades identificadas; y herramientas de automatización de pruebas, como Selenium y Appium, que permiten automatizar las pruebas de seguridad y asegurar que las vulnerabilidades se identifiquen y corrijan de manera oportuna. Estas herramientas son fundamentales para integrar la seguridad en el ciclo de vida del desarrollo de software y garantizar que las aplicaciones sean seguras y confiables (Plant, et al., 2022).

Según indican Melgar, et al. (2021), para implementar esta práctica, se utilizan diversas herramientas como OWASP Software Assurance Maturity Model (Owasp SAMM) para la fase de Control de Calidad, Jenkins para la integración continua, Ansible para la automatización de la infraestructura, Docker para la creación de contenedores y Kubernetes para la orquestación de contenedores. Además, se utilizan herramientas de monitoreo y análisis de logs, como ELK Stack, y herramientas de gestión de vulnerabilidades, como Nessus. Estas herramientas permiten una gestión eficiente y segura de los proyectos en Centros de Datos.

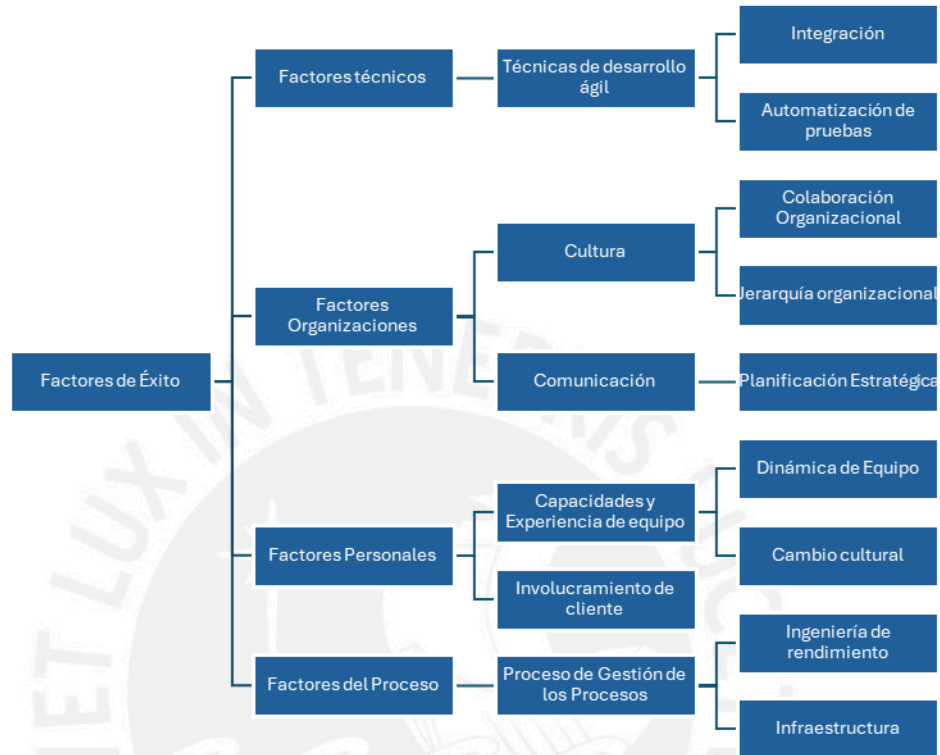
4.1.6. Factores de éxito de los métodos de desarrollo de software

Independientemente de la metodología a usarse, sea ágil o tradicional, existen factores comunes que impactan en los procesos de desarrollo de software en cualquier organización. Y estos comienzan por la claridad del alcance, la estimación realista de los tiempos y del presupuesto, las competencias de los líderes del proyecto, la participación y compromiso de los interesados, y la comunicación y colaboración entre miembros (Muhammad, et al., 2021).

Sin embargo, sólo bastan unos cuantos factores para afectar el desempeño de los métodos ágiles, al tener principios y valores fuertemente enlazados, cualquier desbalance puede ocasionar una mediana contribución al objetivo final (Damyanov, et al., 2024). Estos pueden agruparse en cuatro grupos como factores técnicos, que incluyen a la estrategia de entrega y las técnicas ágiles usadas; a factores organizacionales como la cultura y las comunicaciones, a factores personales como las capacidades y experiencia de los equipos, el involucramiento del cliente y el compromiso de los líderes; y a factores relacionados al proceso tales como el la madurez y nivel de estandarización (Azad & Hyrynsalmi, 2023). Una taxonomía de dichos factores se encuentra ilustrados en la Figura 9.

Figura 9

Taxonomía de factores de éxito del desarrollo ágil.



Nota. Adaptado de DevOps critical success factors—A systematic literature review. Information and Software Technology (2023).

Aportes relevantes como el de Azad & Hyrynsalmi (2023) identifican otros factores adicionales para métodos más específicos como DevSecOps, tales como la jerarquía organizacional, que siendo muy vertical puede desfavorecer las comunicaciones a todo nivel; o el planeamiento estratégico que debe incluir la adopción de prácticas ágiles en las áreas operativas y de soporte, así como el aporte de la infraestructura y los microservicios como parte de los factores técnicos. Estos se aprecian mejor en la siguiente tabla 23.

Tabla 23*Categorías de los factores críticos de éxito*

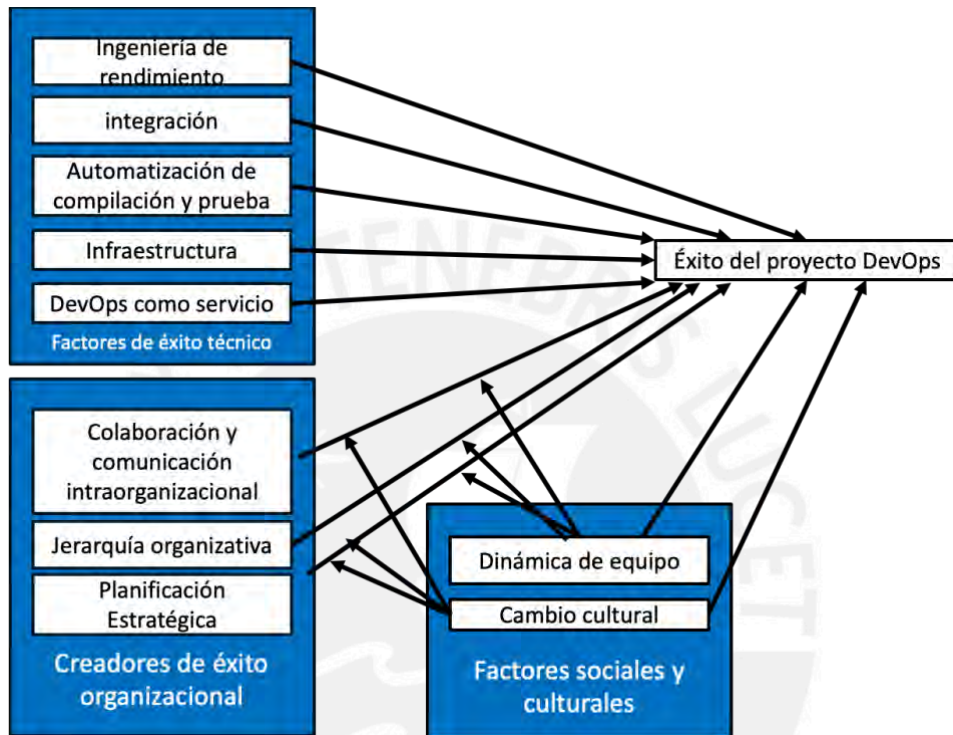
<i>Categoría</i>	<i>Factor crítico de éxito</i>	<i>Ejemplos</i>
Técnica	Ingeniería del desempeño	Complejidad de la ingeniería de rendimiento, efectos de regresión de rendimiento, actualizaciones continuas.
	Integración	Integración con herramientas existentes, mecanismos para la retroalimentación de calidad, proceso de integración continua.
	Automatización de la construcción y de las pruebas	Automatización de desarrollo y pruebas, verificaciones
	Infraestructura	Aprovisionamiento de el despliegue y la infraestructura, diseño de infraestructura común, adopción de sistemas heredados.
Organizacional	DevSecOps como un servicio y retos asociados	
	Organizacional interna, colaboración y comunicación	Colaboración entre el desarrollo y la operación, alineamiento de áreas, colaboración para roles diferentes
	Jerarquía organizacional	Disposición de nuevos roles, soporte a los equipos por encima de jerarquías, colaboración entre áreas
Cultural y social	Planeamiento estratégico	Planificación de la próxima mejora, desarrollo diferente, capacidad de respuesta de calidad a las necesidades comerciales.
	Dinámicas del equipo	Equipos y procesos eficientes, espacios para discutir desacuerdos, resistencia al cambio
	Cambio cultural	Diversidad adoptada, conocimiento adoptado, equipo multidisciplinario, cambios en la cultura de la organización

Nota. Adaptado de *DevOps critical success factors — A systematic literature review*. Azad & Hyrynsalmi (2023). *Information and Software Technology*. Elsevier

En la Figura 10, se muestra la interrelación de los factores de éxito de los métodos ágiles, especialmente en DevOps

Figura 10

Síntesis de los factores críticos de éxito



Nota. Tomado de *DevOps critical success factors — A systematic literature review*. Azad & Hyrynsalmi (2023)

4.1.7. Casos de éxito implementando métodos ágiles

El siguiente caso corresponde a un reporte publicado por Microfocus (2022), correspondiente a la implementación de DevOps como marco de referencia para el desarrollo de software. Desde el inicio de sus operaciones Nationwide Mutual Insurance Co. inició como una pequeña aseguradora de autos, desarrollándose hasta convertirse en una de las compañías de seguros y servicios financieros más grandes del mundo, con más de USD135 mil millones en activos estatutarios, y cuenta con más de 5000 empleados de TI. Nationwide

necesitaba optimizar su capacidad de respuesta ante los requisitos y regulaciones en constante cambio del mercado, aumentar la velocidad de comercialización de nuevos productos y servicios. Para lograr este objetivo Nationwide en colaboración IBM, inició el proceso de transformación ágil y enfocó su desarrollo al desarrollo y entrega continua.

El proceso de transformación considero cambiar la cultura de desarrollo en cascada, la cual era utilizada por Nationwide para desarrollar productos de software que sus miembros, clientes y agentes utilizan en línea, sin embargo, existían brechas entre el software requerido y la velocidad de entrega que era causado por el modelo en cascada. La solución de esta situación, a través de la implementación de la cultura ágil, incluyó la implementación de plataformas de integración y entrega continuas (CI/CD) automatizando el proceso de construcción, prueba y despliegue de software.

Como resultado Nationwide mejoró la calidad de su código en un 50% en un periodo de tres años, se redujo la inactividad de usuarios en un 70% y se fortaleció a sus equipos en aspectos de productividad en un 58%. Datos validados en el reporte de IBM Corporation.

Otro caso singular representa Vodafone en el caso redactado por Microsoft Stories (2023). Vodafone, una destacada empresa de telecomunicaciones en el Reino Unido, llevó a cabo una exitosa transformación digital al adoptar la plataforma de servicios en la nube de Microsoft Azure y utilizar Azure DevOps para mejorar sus procesos de desarrollo y entrega de software. La migración de sus sistemas y aplicaciones a Azure permitió a Vodafone simplificar su infraestructura y reducir costos asociados con la gestión de la misma, mientras que Azure DevOps les brindó una solución integral para la colaboración, planificación, desarrollo, pruebas y entrega de software de manera ágil.

La combinación de Azure y Azure DevOps permitió a Vodafone mejorar la eficiencia de sus equipos de desarrollo al ofrecer herramientas integradas para la planificación de proyectos, seguimiento de tareas, control de versiones y gestión de pruebas. Con Azure DevOps, Vodafone pudo adoptar prácticas de desarrollo ágiles y DevOps, lo que les permitió acelerar la entrega de software y reducir los tiempos de lanzamiento al mercado. Además, la integración y la entrega continua (CI/CD) se volvieron más fluidas y automatizadas, mejorando la calidad y confiabilidad de sus servicios. La implementación de Azure DevOps también permitió a Vodafone mejorar la colaboración entre equipos, facilitando la comunicación y la compartición de conocimientos en un entorno unificado. Esto fomentó la eficiencia y la colaboración efectiva en el desarrollo y entrega de software, lo que resultó en una mayor productividad y una mejora en la calidad de los productos y servicios ofrecidos por Vodafone.

En resumen, la adopción conjunta de Azure y Azure DevOps permitió a Vodafone Telecomunicaciones lograr una transformación digital exitosa al simplificar su infraestructura, reducir costos y mejorar sus procesos de desarrollo y entrega de software. La integración de Azure DevOps en su flujo de trabajo de desarrollo les permitió adoptar prácticas ágiles y DevOps, acelerando la entrega de software, mejorando la calidad y fomentando una colaboración más efectiva entre los equipos. Esto representó un incremento de 25% en las interacciones de usuarios desde el 2018 hasta el 2022, un incremento en la productividad de los desarrolladores de 400% y un incremento de lanzamientos desde 5 veces por año en el 2018 a 500 por mes en el 2022.

4.2. Casos Benchmarks relacionados

La siguiente tabla 24 y tabla 25, resume la comparación en ventajas y desventajas entre los principales métodos ágiles, considerando que DevSecOps ha sido ampliamente discutido y para fines de considerar métodos híbridos, se muestra a continuación.

Tabla 24

Ventajas y desventajas de métodos ágiles (parte 1)

<i>Método Ágil</i>	<i>Ventajas</i>	<i>Desventajas</i>
Feature Driven Development (FDD)	<ul style="list-style-type: none"> - Altamente adaptativa, acepta grandes y tardíos cambios por el cliente. - Entrega de alta calidad después de cada fase. - Resultados de iteraciones entregados en una o cuatro semanas, obteniendo retroalimentación rápida de los usuarios. 	<ul style="list-style-type: none"> - Falta de guías sobre la recopilación de requisitos, análisis y gestión de riesgos. - Requiere un equipo experto con altas habilidades en diseño y gestión de riesgos. - No presta suficiente atención a los temas críticos del proyecto.
Test Driven Development (TDD)	<ul style="list-style-type: none"> - Mejor manejo de los efectos de la prueba al realizarse simultáneamente con el desarrollo. - Mejora de la calidad del código al probarse todos los códigos durante el desarrollo. - Desarrollo de pequeñas funcionalidades en cada iteración reduce la complejidad del producto final. 	<ul style="list-style-type: none"> - Necesidad de habilidades adicionales en redacción de casos de prueba. - Dificultades en sincronización entre funcionalidades y partes. - Poca documentación generada, concentrada en los casos de prueba. - Alto consumo de tiempo en fallas repetitivas. - Enfoque únicamente en actividades de desarrollo, sin principios de gestión en desarrollo de software.
Scrum	<ul style="list-style-type: none"> - División del producto final en partes manejables y entendibles, aumentando el conocimiento compartido y la comunicación. - Visibilidad sobre toda la etapa, incluida retroalimentación del cliente. - Compartición de responsabilidades en todo el equipo. - Permite autoevaluaciones en cada sprint. - Proceso simple. - Posibilidad de incorporar elementos ignorados en etapas del proyecto. 	<ul style="list-style-type: none"> - Posibles violaciones a acuerdos al no existir responsabilidades individuales bien definidas. - Falta de prescripción de prácticas específicas, métodos o guías de ingeniería.

Nota. Tomado de A Review Of Agile Methodologies In Product Lifecycle Management: Bridging Theory And Practice For Enhanced Digital Technology Integration. Ibeh, C. V., Awonuga, K. F., Okoli, U. I., Ike, C. U., Ndubuisi, N. L., & Obaigbena, A. (2024).

Tabla 25*Ventajas y desventajas de métodos ágiles (parte 2)*

<i>Método Ágil</i>	<i>Ventajas</i>	<i>Desventajas</i>
Extreme Programming (XP)	<ul style="list-style-type: none"> - Desarrollo incremental apoyado por entregas pequeñas y frecuentes. - Aceleración de desarrollos si pasa la prueba. - Mantenimiento de simplicidad mediante refactorización del código. - Mejora de calidad mediante desarrollo de pruebas automatizadas antes de integrar nuevas características. 	<ul style="list-style-type: none"> - Baja capacidad de soporte de otros equipos, centrado en la comunidad. - Dependencia de documentación informal para mantener detalles importantes del proyecto. - Práctica de involucramiento del cliente muy demandante en tiempo y esfuerzo.

Nota. Tomado de A Review Of Agile Methodologies In Product Lifecycle Management:

Bridging Theory And Practice For Enhanced Digital Technology Integration. Ibeh, C. V., Awonuga, K. F., Okoli, U. I., Ike, C. U., Ndubuisi, N. L., & Obaigbena, A. (2024).

Una característica común que mantienen estas metodologías es su atención en el cliente. Cada una de ellas será adaptada a una determinada característica del proyecto y de la situación de su entorno. La siguiente tabla 26 y tabla 27 se muestra herramientas o métodos aplicados en cada una de las metodologías indicadas.

Tabla 26*Prácticas de métodos ágiles más populares (parte 1)*

<i>Método ágil</i>	<i>Prácticas ágiles</i>
Scrum	Sprint, Planificación, Standups diarios, Retrospectivas, Product Owner, Scrum Master, etc.
Kanban	Visualización, Limitar el trabajo en curso, Ciclos de retroalimentación, etc.
Extreme Programming XP	Compilaciones regulares de software, integración continua, desarrollo basado en pruebas, programación en pares, etc.
FDD	Equipos de características, inspecciones de código, propiedad del código, compilaciones regulares del software, etc.
Crystal	Entrega frecuente, mejora reflexiva, Comunicación cercana. Alta participación del usuario, etc.

Nota. Tomado de *Visión general de los métodos ágiles más comunes y las prácticas ágiles asociadas* (Shastri, 2021; Hossain, 2023)

Tabla 27*Prácticas de métodos ágiles más populares (parte 2)*

<i>Método ágil</i>	<i>Prácticas ágiles</i>
Scaled Agile Framework (SAFe)	Planificación de incremento de productos, evento de inspección y adaptación, tren de lanzamiento ágil, etc.
Large Scale Scrum (LeSS)	Retrospectivas generales, Sprint Planning 1 y 2, Lean Thinking, etc.
DevSecOps	Integración de Políticas de gobernanza y seguridad al ciclo de vida de software, gestión de incidencias, gestión de vulnerabilidades, implementación de herramientas automatizadas (pruebas vulnerabilidades), entrega continua.

Nota. Tomado de Visión general de los métodos ágiles más comunes y las prácticas ágiles asociadas (Shastri, 2021; Hossain, 2023)

Haciendo uso de la entrega continua, Microsoft – Customer Stories (2022) muestra un caso de negocio interesante, basado en las herramientas de despliegue de DevOps. Los Split Software y J.J. Keller se unieron para impulsar el cumplimiento de la seguridad mediante el uso de Microsoft Azure y Azure DevOps. J.J. Keller, líder en el cumplimiento de regulaciones de seguridad en los Estados Unidos, buscaba una forma más eficiente y escalable de ofrecer sus servicios a los clientes. Mediante la migración a Azure y la adopción de Azure DevOps, lograron mejorar la entrega de servicios de cumplimiento de seguridad.

Azure DevOps fue fundamental en este proceso, ya que proporcionó a J.J. Keller herramientas integrales para la colaboración, planificación, desarrollo y entrega continua de software. Gracias a Azure DevOps, J.J. Keller pudo adoptar prácticas de desarrollo ágiles y DevOps, acelerando la entrega de software y mejorando la calidad de sus servicios de cumplimiento de seguridad. Además, Azure DevOps permitió una mayor eficiencia y colaboración entre los equipos de desarrollo y operaciones de J.J. Keller. La combinación de Azure y Azure DevOps permitió a J.J. Keller escalar rápidamente sus servicios de cumplimiento de seguridad según la demanda. La flexibilidad y escalabilidad de Azure les permitieron adaptarse a las necesidades cambiantes de sus clientes, mientras que Azure

DevOps agilizó los procesos de desarrollo y entrega de software. La integración continua y la entrega continua (CI/CD) se volvieron más eficientes y automatizadas, lo que mejoró la calidad y confiabilidad de los servicios de cumplimiento de seguridad de J.J. Keller.

En resumen, la adopción de Microsoft Azure y Azure DevOps permitió a J.J. Keller mejorar la entrega de servicios de cumplimiento de seguridad. Azure DevOps proporcionó herramientas y prácticas ágiles y DevOps para acelerar la entrega de software y mejorar la colaboración entre equipos. La escalabilidad y flexibilidad de Azure permitieron a J.J. Keller adaptarse rápidamente a las necesidades del mercado, asegurando la calidad y confiabilidad de sus servicios de cumplimiento de seguridad. Esto permitió obtener una mejora representada en tiempo de despliegues, de 4 horas a 30 minutos después de la adopción de DevOps, reduciendo costos relacionados al mantenimiento.

Conclusiones

El desarrollo de software en la actualidad no puede estar exento de un marco referencial que le permita manejar la complejidad del entorno. Cada metodología representa una determinada forma de enfrentar dicha complejidad y es necesario entender la aplicabilidad de cada una de ellas en determinadas circunstancias y realidades tanto al interior como al exterior de las organizaciones. DevSecOps es vigente y oportuno dada la variedad de amenazas y requerimientos cada vez mayores de protección de la información, representa una buena forma de abordar los proyectos considerando factores de seguridad desde el desarrollo del código.

Las metodologías ágiles demuestran su valor reemplazando a metodologías tradicionales colocando énfasis en los despliegues y en la entrega y calidad de los entregables; sin embargo, existen oportunidades en que los métodos y prácticas puedan ser

adaptados en cada organización, incluso hasta una combinación de ellos generando métodos híbridos entre tradicionales y ágiles inclusive. El juicio de los líderes de proyecto para asignar el correcto marco referencial es tan importante como el marco en sí.



Capítulo V: Análisis del problema / Oportunidad detectada

En este capítulo se analiza el método utilizado para la identificación de los problemas en el área de soporte técnico e investigación y desarrollo, así como el análisis de estos en MS4M. Es importante considerar para esto, un análisis detallado de las causas a los problemas de la empresa en las áreas de soporte y desarrollo (inversión y desarrollo), correspondiente a los productos y su impacto en los clientes.

5.1. Causas identificadas

Estando identificados los problemas relacionados a la alta demanda de soporte tecnológico dentro de los clientes de MS4M, se procedió a realizar una encuesta sobre las posibles causas que sostienen esta alta demanda entre el personal del área de Soporte Tecnológico, y en forma separada también con el personal del área de Investigación y Desarrollo. El resultado de esta actividad se detalla a través del diagrama causa y efecto de la Figura 11.

Figura 11

Diagrama de causa y efecto de resultados de encuesta de percepción del personal sobre altos costos del soporte del nivel 3.



Nota. Tomado de Encuesta sobre la percepción de causas de alta demanda de soporte técnico de los productos de MS4M en el área de Investigación y Desarrollo. Equipo de Investigación (2023).

En lo que respecta a la metodología, un 66.7% del personal en el equipo de MS4M percibe que, aunque la organización dispone de metodologías de desarrollo, estas no se aplican correctamente. Otro hallazgo relevante de la encuesta reveló que el 20% de los encuestados no tiene conocimiento de las metodologías utilizadas en la organización. En cuanto a la autoevaluación del personal, el 22.2% de los evaluados percibe que el equipo no posee habilidades técnicas suficientes. De estos, el 15.6% señala que se necesita una mayor experiencia en el campo, mientras que el 42.2% opina que el personal de desarrollo no es adecuado para llevar a cabo sus actividades.

En relación con los materiales y equipos utilizados para llevar a cabo las actividades de soporte, el 40% de los evaluados percibe que el software actualmente disponible no es suficiente, mientras que el 17.8% considera que el hardware destinado para este propósito no es adecuado. Sin embargo, un 4.4% de los evaluados percibe que no existe software especializado. Asimismo, respecto a los indicadores, el 60% de los evaluados expresó que los indicadores de calidad de los productos son insuficientes o inadecuados. Además, un 26.7% sostiene que no existen indicadores de medición de calidad disponibles. En un aspecto más preocupante, el 2.2% de los encuestados desconoce la existencia de indicadores.

El 71.1% percibe que la infraestructura de soporte es insuficiente, mientras que el 4.4% señala que no existe infraestructura para esta actividad. Por otro lado, en la gestión del servicio de soporte, se revela que el 6.7% de los encuestados señala que no existe una planificación de desarrollo de productos, el 22.2% de estos percibe que la planificación de proyectos de desarrollo de productos es inadecuada, sin embargo, el 64.4% indicó que el principal factor que afecta la gestión del servicio se debe a la alta demanda del negocio.

La ficha de especificación de la encuesta realizadas, así como el detalle de los resultados de la encuesta aplicada al personal de MS4M se encuentran en el Anexo I, permitiendo destacar la importancia crítica de abordar y tomar medidas con respecto a la metodología de desarrollo utilizada en la organización. La percepción generalizada de que las metodologías disponibles no se aplican correctamente y la falta de conocimiento de algunas metodologías entre un segmento de los encuestados indican una brecha en la implementación y la comunicación de las mejores prácticas de desarrollo. Por lo tanto, se considera prioritario reforzar y estandarizar los procesos de desarrollo en toda la organización, asegurando su efectiva implementación y promoviendo la comprensión generalizada entre todo el personal

de soporte. Este enfoque busca mejorar la calidad de los productos y aumentar la eficiencia del equipo de trabajo.

MS4M registra las atenciones del soporte que brinda a sus clientes a través del software *JIRA SERVICE DESK*, mediante en forma complementaria a un catálogo de servicios vigente le permite clasificar sus atenciones para fines de retroalimentación a su área de desarrollo de productos, se cuentan bases de datos que pertenecen a las plataformas de *ATLASSIAN* tales como *JIRA SERVICE DESK*, *CONFLUENCE*, *BITBUCKET* y *JIRA SOFTWARE*. Dichas plataformas se integran en diversos procesos, sobre todo en el relacionado al soporte tecnológico, desde donde permite el análisis de datos mediante una clasificación desde su catálogo de servicios. El proceso se muestra en la Figura 12.

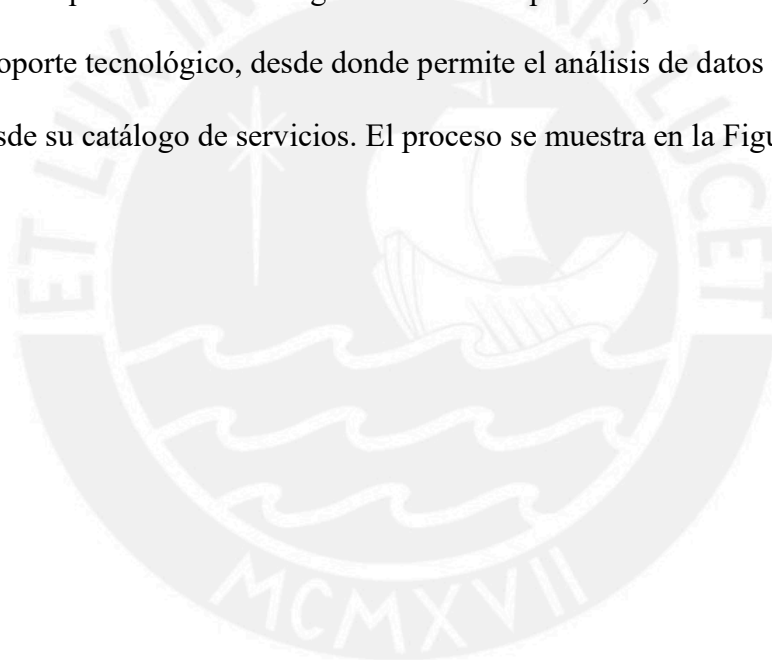
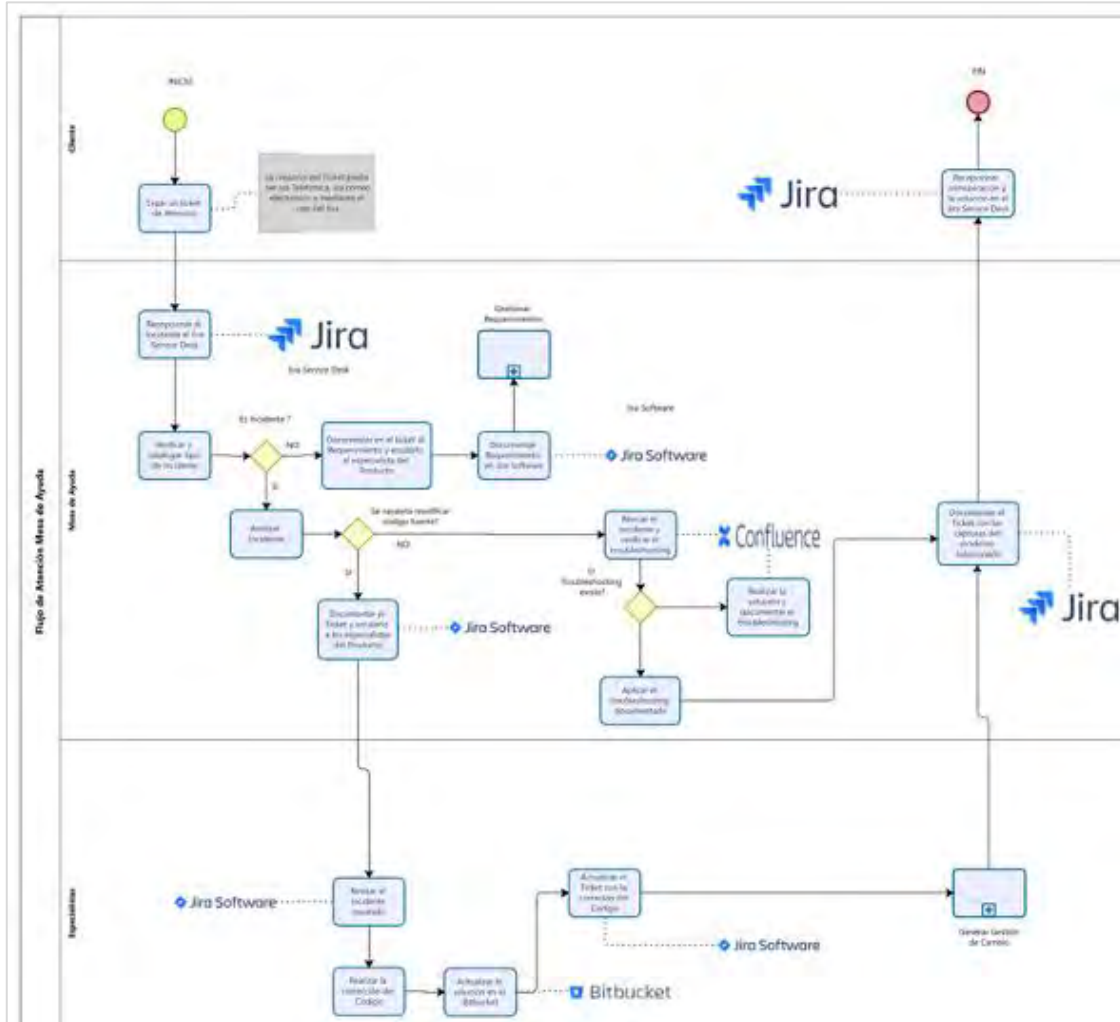


Figura 12

Diagrama de proceso de soporte



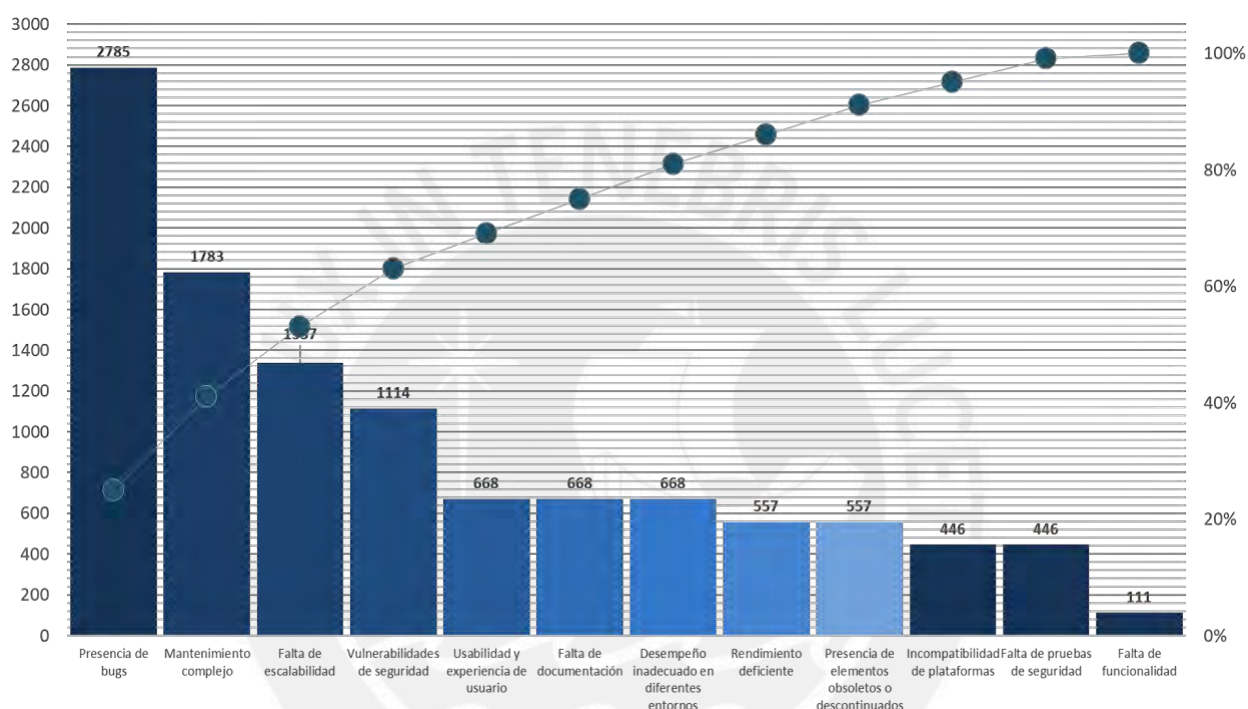
Nota. Tomado de Mapa de Procesos de Soporte de MS4M. Equipo de Investigación (2023).

La información relevante a la casualidad de los eventos de nivel 1 y 3, que demandan la totalidad de recursos del área de soporte y el 17.7% de los recursos del área de investigación y desarrollo, determinan que entre enero y junio de 2023, se registraron un total de 5,713 incidentes a ser resueltos, de los cuales 18% son escalados a nivel 3 según se muestra en la tabla 6 y tabla 7. Dicha cantidad de eventos determina también la siguiente

clasificación sobre la causa en el producto a ser evaluado, según la versión actualizada del catálogo de servicios de MS4M. Los resultados se muestran en la Figura 13.

Figura 13

Diagrama de Pareto de causas de incidencias relacionados al producto.

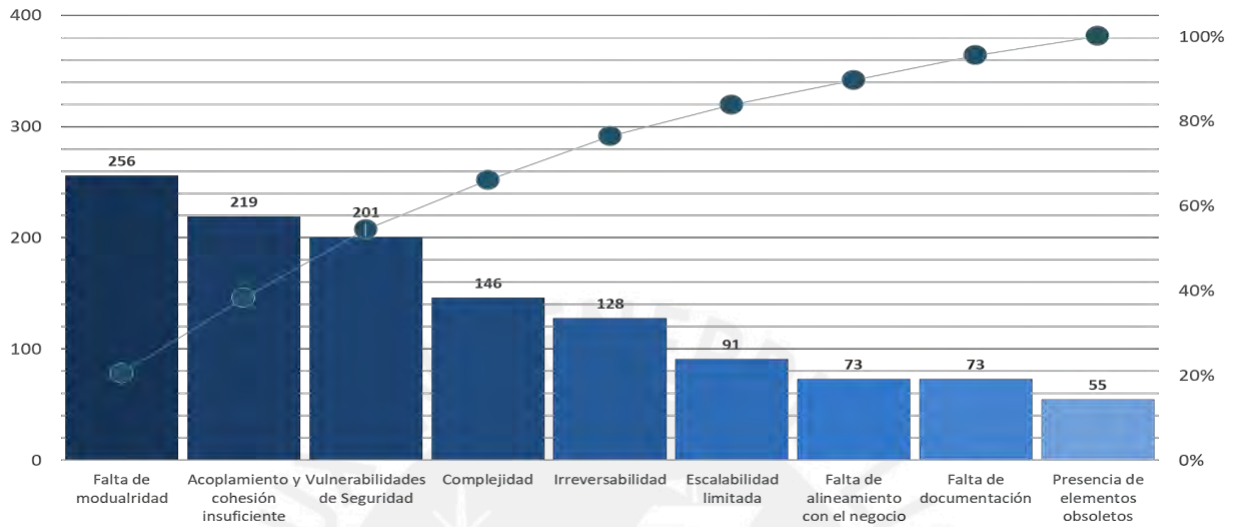


Nota. Elaboración propia desde base de datos de JIRA SERVICE DESK de MS4M. Equipo de Investigación (2023).

Siendo la presencia de errores, complejidad seguridad y escalabilidad los principales campos a ser investigados en los productos de MS4M, guardan relación directa a la arquitectura, la calidad de código y al despliegue de los productos, por ello es que la segunda dimensión a ser analizada corresponde a las características de arquitectura y de código que se muestran en la Figura 14 y 15.

Figura 14

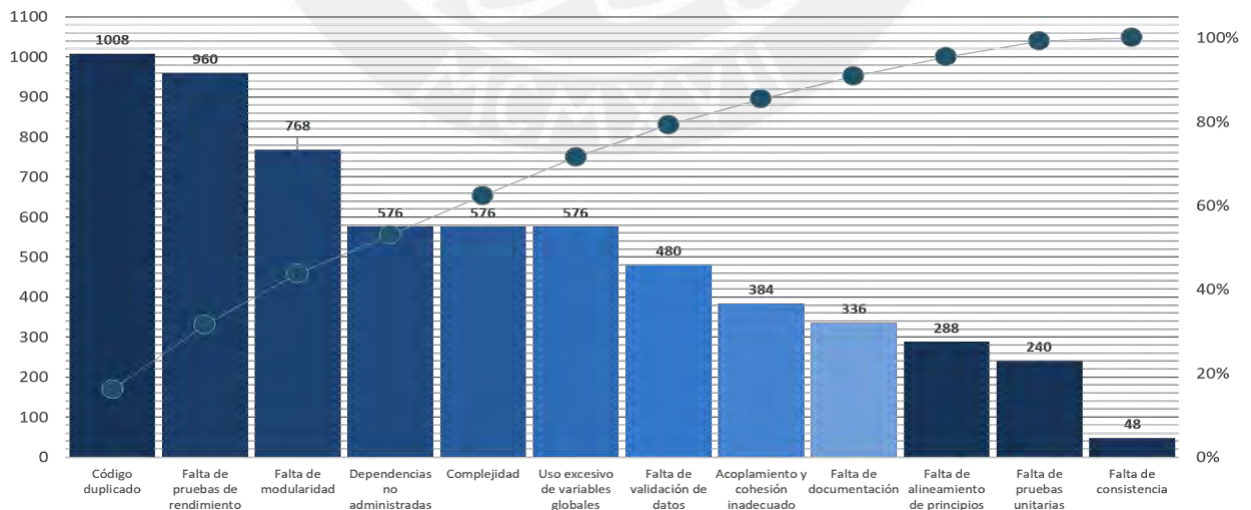
Diagrama de Pareto de causas de incidencias relacionados a la arquitectura



Nota. Elaboración propia desde base de datos de JIRA SERVICE DESK de MS4M. Equipo de Investigación (2023).

Figura 15

Diagrama de Pareto de causas de incidencias relacionados a calidad de código



Nota. Elaboración propia desde base de datos de JIRA SERVICE DESK de MS4M. Equipo de Investigación (2023).

Las figuras 14 y 15 determinan que efectivamente, la modularidad tanto en la arquitectura como en el código, duplicidad y complejidad, así como lo relacionado al despliegue e implementación representan aspectos a ser tratados en las evoluciones del producto, en sus nuevas versiones. La diferencia de la cantidad total de tickets generados en el periodo versus el total de las clasificaciones de causas obedece a que un solo ticket puede tener más de una causa asociada.

5.2. Matriz de Priorización Causa-Raíz del Problema / Factibilidad de oportunidad detectada

Luego de haber identificado los problemas relacionados al producto, la arquitectura y al código en MS4M, elaboraremos la selección de priorización a través de la matriz de Vester, que fue desarrollada por Fredrik Vester en la década de 1970, que se usa como una herramienta de la gestión empresarial y en la toma de decisiones estratégicas. Esta matriz, diseñada para abordar problemas complejos y sistémicos, confronta las alternativas disponibles entre ellas, midiendo su influencia y dependencia sobre las otras, generando cuatro zonas donde se clasifican entre elementos neutros, críticos, activos, pasivos e indiferentes (Jodlbauer, et al., 2022).

Los problemas identificados se presentan a continuación en la siguiente tabla:

Tabla 28*Prácticas de métodos ágiles más populares*

<i>Ítem</i>	<i>Problema</i>
P1	Presencia de bug
P2	Vulnerabilidad de Seguridad.
P3	Falta de escalabilidad
P4	Falta de alineamiento con el negocio
P5	Código duplicado.
P6	Uso excesivo de variables globales
P7	Falta de lineamientos con los principios

Nota. Descripción de los problemas

A continuación, se detallarán los problemas identificados:

Presencia de bug. - Se refiere a la existencia de un error o fallo en un programa de software. Un bug es un término coloquial utilizado en el ámbito de la programación informática para describir un comportamiento no deseado o incorrecto en un programa.

Vulnerabilidad de seguridad. - Se refiere a una debilidad en el software que puede ser aprovechada por un atacante para comprometer la seguridad del sistema.

Falta de escalabilidad. - Se refiere a la incapacidad de un sistema, aplicación o arquitectura para crecer o manejar un aumento en la carga de trabajo de manera eficiente.

Falta de alineamiento con el negocio. - Se refiere a la situación en la que las actividades, objetivos y resultados de un área o proyecto no se encuentran alineados con los objetivos y metas más amplios de la organización o empresa.

Código duplicado. - Se refiere a la existencia de fragmentos de código que son iguales o muy parecidos en diferentes partes de un programa o sistema de software.

Uso excesivo de variables globales. - El uso excesivo de variables globales en programación generalmente es considerado una mala práctica y puede hacer que el código sea más complejo y difícil de entender.

Falta de alineamiento de principios de desarrollo de software. - Se refiere a la situación en la que los equipos de desarrollo, los procesos y las prácticas adoptadas en un proyecto no están en sintonía con los principios fundamentales del desarrollo de software. Esto puede conducir a problemas significativos en términos de eficiencia, calidad del código, mantenibilidad y éxito general del proyecto.

Análisis de los problemas

Los problemas previamente detectados se analizarán en función de cómo se afectan mutuamente, considerando la aplicación de criterios específicos y los valores numéricos que se pueden asignar a las relaciones de los problemas en la matriz de Vester. En la tabla 29 se establece los valores de los criterios para realizar el análisis comparativo posterior:

Tabla 29*Criterios para evaluar la influencia entre los problemas*

<i>Valor</i>	<i>Descripción</i>
0	No existe relación entre problemas
1	Existe una relación débil entre los problemas
2	Existe una relación moderada entre los problemas
3	Existe una relación fuerte entre los problemas

Nota. Elaboración propia.

Los problemas se comparan unos a otros para determinar su nivel de influencia, la influencia de cada problema se mide en dos dimensiones: activo (causal) y pasivo (consecuente). X representa la influencia activa, es decir, el impacto que el problema tiene sobre otros. Y representa la influencia pasiva, es decir, el impacto que otros problemas tienen sobre el problema. Bajo los criterios y puntajes donde van desde cero hasta tres, se califica la nula, la leve, la mediana y la alta influencia entre todas respectivamente, una a una. De esa forma se ha obtenido los resultados mostrados en las Figuras 16 y 17.

Figura 16

Resultados de análisis causas-efecto aplicando Matriz de Priorización de Veste.

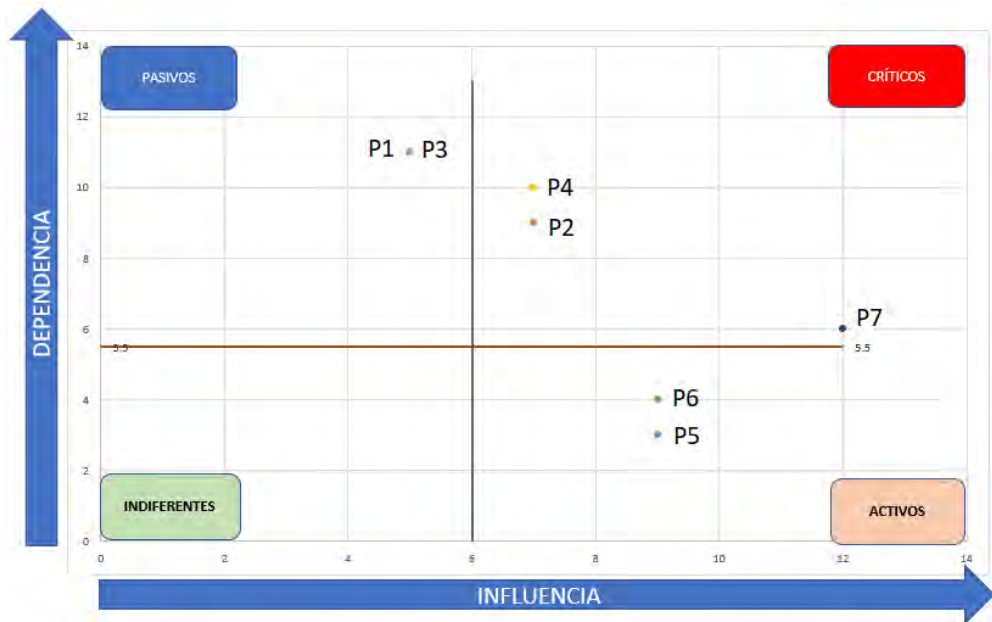
Análisis de Causa - Efecto		P1	P2	P3	P4	P5	P6	P7	EJE X (INFLUENCIA)
		Presencia de bugs	Vulnerabilidades de seguridad	Falta de escalabilidad	Falta de alineamiento con el negocio	Código duplicado	Uso excesivo de variables globales	Falta de alineamiento de principios de desarrollo de software	
P1	Presencia de bugs	2	2	1	2	0	0	0	5
P2	Vulnerabilidades de seguridad	2	2	2	3	0	0	0	7
P3	Falta de escalabilidad	1	0	2	2	0	0	2	5
P4	Falta de alineamiento con el negocio	2	1	2	2	1	1	0	7
P5	Código duplicado	2	2	2	0	2	1	2	9
P6	Uso excesivo de variables globales	2	2	2	1	0	2	2	9
P7	Falta de alineamiento de principios de desarrollo de software	2	2	2	2	2	2	2	12
EJE Y (DEPENDENCIA)		11	9	11	10	3	4	6	

Nota. Elaboración propia. Base de datos de JIRA SERVICE DESK de MS4M. Equipo de Investigación (2023).

Los datos de la matriz de Vester se mostrarán en un plano cartesiano, el cual se divide en cuatro cuadrantes, distribuyéndose en el primer cuadrante los problemas críticos, segundo cuadrante los problemas pasivos, tercer cuadrante problemas indiferentes y en el cuarto cuadrante los problemas activos.

Figura 17

Resultados de priorización de causas-efecto en cuadrantes de Matriz de Vester



Nota. Elaboración propia.

La discusión de resultados nos lleva a la siguiente conclusión. MS4M debe priorizar los problemas relacionados a la calidad de su código en cuanto al alineamiento con los principios básicos de generación de software y vulnerabilidades de seguridad. Al presentarse en una zona crítica se estima que tienen alta influencia sobre otros problemas existentes y potenciales, por lo que el esfuerzo demanda una mayor cantidad de recursos y su impacto es mayor.

De igual forma, la duplicidad de código tiene una alta influencia sobre otros problemas existentes pero una menor dependencia de otros, por lo que definirá una segunda prioridad.

Capítulo VI: Alternativas de solución evaluadas

En el presente capítulo se realizará la ponderación y valoración de las alternativas de solución, utilizando la metodología de valoración por jerarquía AHP, considerando los beneficios esperados y los desafíos que conlleva determinada solución.

Habiendo identificado las causas asociadas a la calidad de código, entendiendo desde la presencia de bugs, arquitectura duplicidad y coherencia, y su impacto en la escalabilidad de los problemas y soluciones técnicas incrementado el costo de este proceso de una forma tal que comprometa los objetivos estratégicos de MS4M, se analizan una serie de alternativas que abordan temas técnicos hacia el proceso de desarrollo de software, hasta opciones de gestión integral de la organización en donde las soluciones a proponer puedan complementarse y sobre todo, sostenerse.

6.1. Alternativas para resolver el problema

Se analizarán a continuación las siguientes alternativas de solución considerando de que no serán excluyentes, y desde su relación con el desarrollo de software, en forma de impactos positivos y con los riesgos que conllevan.

6.1.1. *Implementar un modelo de gobernanza organizacional*

Descripción. La gobernanza organizacional es el conjunto de prácticas y procesos que se utilizan para dirigir y gestionar una organización, y se enfoca principalmente en la toma de decisiones, la supervisión y la rendición de cuentas en una empresa, sobre todo cuando se trata de lidiar con los cambios y complejidades propias de la innovación (Jacobides, et al., 2024). (Chen, et al., 2021) indican que el gobierno de las organizaciones se establece en dimensiones tales como la asignación de recursos, información, autonomía, reconocimientos, acceso, resultados, comportamientos y relaciones externas que, en conjunto, designan los mecanismos que controlan las actividades principales de la organización.

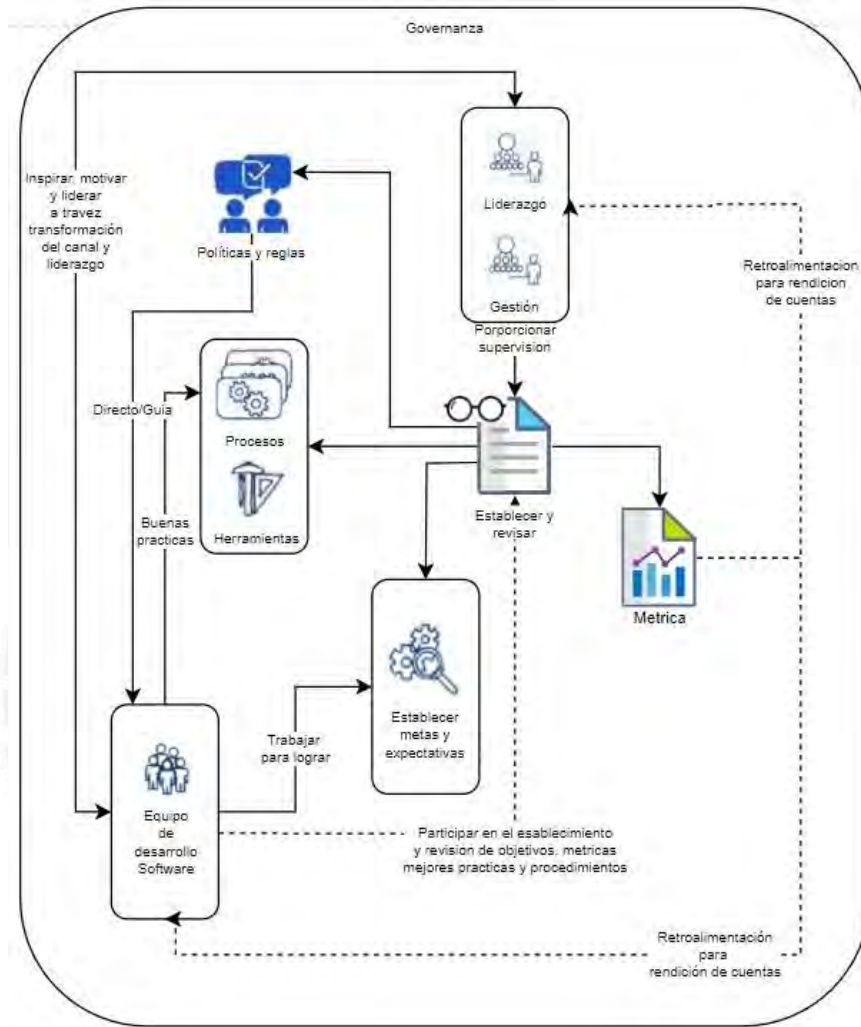
Beneficios esperados. La gobernanza en empresas que desarrollan software puede aportar cinco aspectos principales, enfocadas precisamente a la calidad del producto final.

La primera está relacionada a la supervisión a través de la retroalimentación de los equipos de trabajo; la segunda está relacionada a la importancia de la rendición de cuentas durante el desarrollo y aprendizaje, la tercera está dirigida al control mediante reuniones donde se discuten las lecciones aprendidas; la cuarta está relacionada a la dirección y alineamiento con los objetivos de la organización y, por último, la definición de procesos guía para mitigar riesgos y facilitar el control. Estos aspectos finalmente redundan en el liderazgo, la gestión y el equipo de desarrollo como críticos para la gobernanza (Tuape, et al., 2022).

Un mejor dimensionamiento del beneficio se expresa en el siguiente modelo representado por Taupe, Ilyambo & Kasurinen (2022) expresados en relaciones de rendición de cuentas, retroalimentación y comunicación interna.

Figura 18

Modelo de rendición de cuentas, retroalimentación y comunicación interna



Nota. Extraído y adaptado de Tuape, M., Iiyambo, P., & Kasurinen, J. (2022). Organizational governance: Resolving insufficient practice and quality expectation in Small Software Companies. In Proceedings of the 2022 European Symposium on Software Engineering

Este modelo se centra en el liderazgo, la gestión y el equipo de desarrollo de software como aspectos críticos de la gobernanza organizacional, en donde la describe como una responsabilidad de los ejecutivos para proporcionar liderazgo, los gerentes de la

organización de la operatividad diaria y el equipo de desarrollo de la creación del valor.

Propone un liderazgo transformacional de manera de inspiración y logro de objetivos comunes con rendición de cuentas y niveles de retroalimentación continuos.

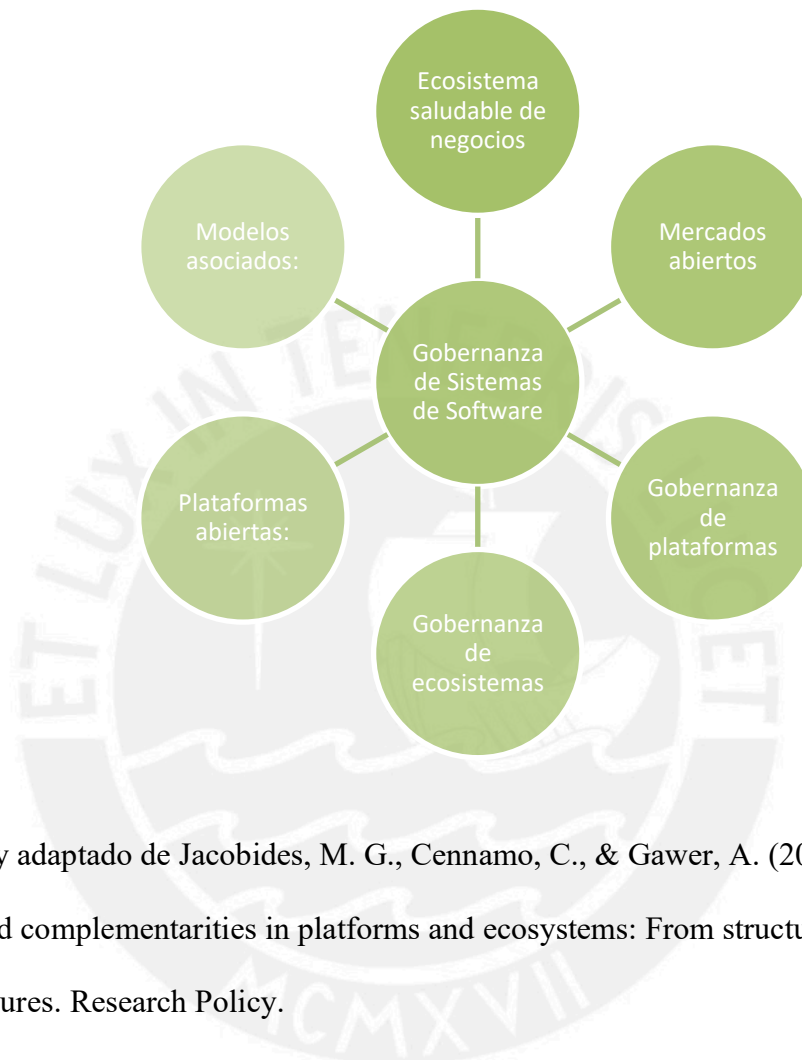
Desafíos. El principal desafío de las políticas de gobernanza en las empresas es su nivel de madurez, y esto se ve reflejado principalmente en la forma en que la empresa toma decisiones, comunica y define objetivos. Los niveles de madurez respecto a los procesos influyen en la medida que soportan las políticas y si su nivel es bajo, traerá problemas de falta de transparencia, falta de colaboración y falta de alineamiento a objetivos, sobre todo, a empresas que desarrollan software.

Existen modelos de madurez de la gobernabilidad de ecosistemas de software como plantea, que se concentran en las facilidades de desarrollo de las áreas de foco, siendo el desarrollo de software una de ellas, como se expresa en las siguientes figuras (Nugroho, et al., 2022). Jacobides, Cennamo & Gawer (2024), plantean que las áreas de enfoque contienen un número de capacidades asociadas, que son las habilidades y competencias necesarias para alcanzar sus objetivos, dichas capacidades tienen un nivel de madurez.

En esta figura, se muestra cómo los modelos de madurez definen donde se encuentran las capacidades, las prácticas y los niveles para ayudar a las organizaciones a mejorar su eficacia en un área de enfoque específica, que puede ser el de desarrollo de software.

Figura 19

Áreas de enfoque em el desarrollo de software



Nota. Extraído y adaptado de Jacobides, M. G., Cennamo, C., & Gawer, A. (2024).

Externalities and complementarities in platforms and ecosystems: From structural solutions to endogenous failures. *Research Policy*.

Tal como se describió en la tabla 21, MS4M cuenta con el 80% de sus procesos en niveles iniciales e incompletos en su diagnóstico de madurez, lo que representa un alto riesgo para la adopción de criterios de gobernanza en cuanto a su sostenibilidad en el tiempo, en el corto plazo. La adopción de criterios de gobernanza aportará mayormente cuando sus procesos se encuentren en un nivel de definición al menos o en su mayoría.

6.1.2. Implementar un modelo de arquitectura empresarial

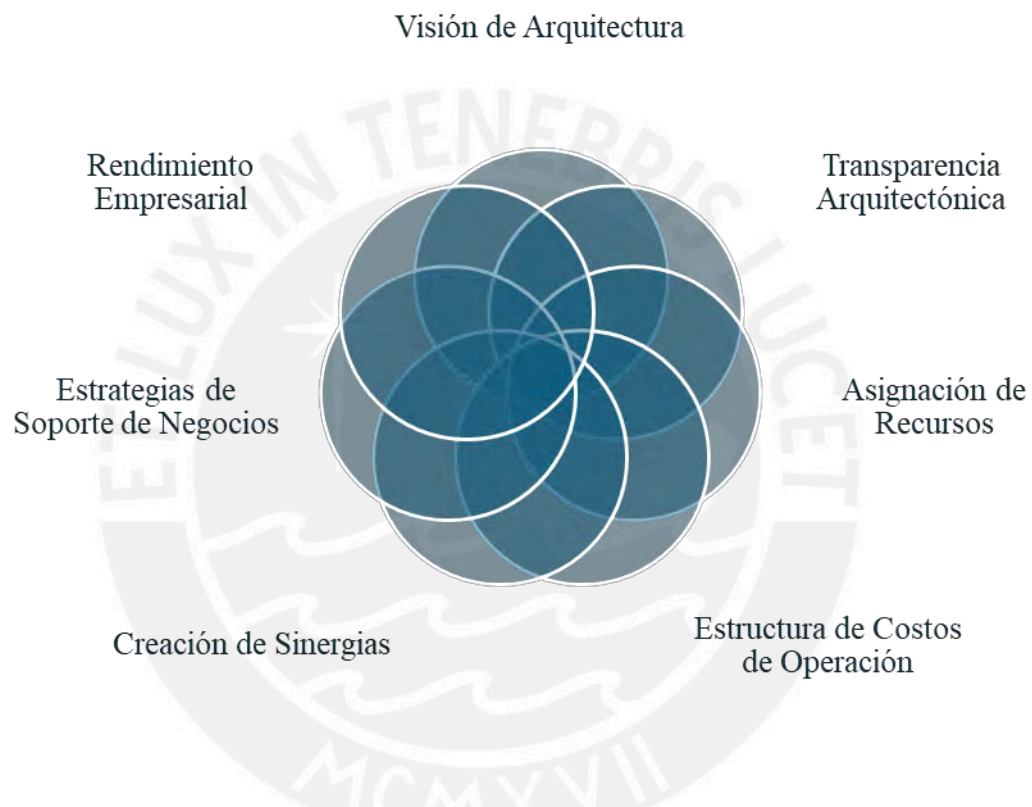
Descripción. La arquitectura empresarial se define como un conjunto coherente de principios, métodos y modelos que se usan en el diseño y la realización de la estructura organizativa, los procesos de negocio, los sistemas de información y la infraestructura de una organización. Tiene como objetivo alinear los objetivos del negocio y la tecnología a usarse, así como ayudar a gestionar, controlar y a evaluar la complejidad de las empresas, sus procesos, sus aplicaciones y su infraestructura, otorgando una visión holística del negocio (Ilin, et al., 2021). Las transiciones hacia un modelo esperado se muestran en la siguiente figura.

Beneficios esperados. La arquitectura empresarial tiene dimensiones que se concentran en el diseño de los procesos, incluyendo al desarrollo de software. Esto incluye además la adopción de marcos de referencia que sirvan de guía para la formalización de procesos, puede proporcionar guías o métodos de trabajo específicos, proporcionar canales de comunicación empresariales que serán usados para las coordinaciones, colaboraciones y retroalimentación en el desarrollo de software, procesos de gestión de cambio para despliegues (Uysal & Mergen, 2021). Para el caso de una empresa que adopta o contiene metodologías ágiles, la arquitectura empresarial puede permitir además alinear el desarrollo a los objetivos estratégicos de la empresa de largo plazo y evitar que se incluyan soluciones de corto plazo que no sean sostenibles en el largo plazo, además, permitir la integración entre diferentes tipos de soluciones cuando sea requerida, permitiendo de esta manera la generación de soluciones aisladas sin capacidad de generar valor conjunto (Brée & Karger, 2022).

En la figura se puede observar el efecto en el valor que produce la implementación de un modelo de arquitectura empresarial.

Figura 20

Beneficios de la implementación de arquitectura empresarial



Nota. Extraído y adaptado de Kurnia, S., Kotusev, S., Shanks, G., Dilnutt, R., Taylor, P., & Milton, S. K. (2021). Enterprise architecture practice under a magnifying glass: linking artifacts, activities, benefits, and blockers.

Desafíos. La calidad de la arquitectura organizacional influye mucho en el resultado de calidad en los procesos. La calidad puede ser definida mediante marcos de referencia como CMQF - Conceptual Model Quality Framework o el Sequal que miden la capacidad de representación de los modelos de arquitectura con los modelos existentes expresados en

procesos y sus relaciones, es decir la medida en la que representa a la empresa y sus componentes.

Un modelo de bajo nivel será incapaz de generar procesos de calidad generando confusión, reprocesos y también puedan existir dificultades de comprender, que los usuarios no logren identificar mejoras o tomen decisiones desinformadas, y puede carecer de detalle y contexto de forma tal que se omitan en etapas importantes de los procesos y generen vacíos de responsabilidades.

Por lo que es muy importante que cuando se emprenda la implementación de modelos de arquitectura, primero se mida su nivel de representación que permita generar calidad, el desafío puede ser demasiado grande para empresas inexpertas y las expectativas sobrevaloradas.

Pero el desafío más importante que deben cumplir los modelos de arquitectura empresarial es la atribución de los logros que obtiene, ya que son difícilmente asignados a sus aportes por que estos se encuentran hasta un tercer nivel de impacto, sobre todo en implementaciones extensas en tiempo y alcance (Brée & Karger, 2022).

Tiempo de implementación. Se recomienda que sea gradual, con una arquitectura básica centrada en los procesos principales que puede tomar entre 9 a 12 meses, sin embargo, los verdaderos aportes serán visibles en un nivel de maduración y evolución de la arquitectura que puede durar de 3 a 5 años (Brée & Karger, 2022).

6.1.3. Implementar metodologías ágiles

Las metodologías ágiles y sus aportes para este caso fueron ampliamente descritas en el capítulo III, sin embargo, existen algunos desafíos que afrontar en empresas como MS4M.

Desafíos. Implementar metodologías ágiles tiene muchos desafíos, para este caso evaluaremos el tratamiento de los requerimientos funcionales y no funcionales presentes en el desarrollo de software, primero porque las metodologías ágiles deben lidiar con ambas, sin embargo, los requerimientos funcionales son más notorios y fáciles de medir, por lo que los requisitos no funcionales deben de ser detallados más cuidadosamente además porque se priorizan las entregas rápidas. Los requisitos no funcionales pueden variar hasta aspectos de usabilidad, seguridad y rendimiento del software (Jarzebowicz & Weichbroth, 2021), por lo que el método ágil como *DevSecOps* debe contemplar todos estos aspectos.

Existen otros desafíos dentro de la implementación de metodologías ágiles en organizaciones, y son del tipo social y cultural. Una empresa de bajo nivel de madurez de comunicaciones, con una escasa arquitectura tecnológica que facilite la coordinación de los equipos, y sobre todo una cultura orientada al resultado requiere de una mayor atención a la gestión del cambio de la forma en que se planifica, ejecuta y entrega el software al cliente interno o externo (Abidin & Samat., 2021) o simplemente en la gestión de documentos, una cultura con alto detalle documentario de requisitos funcionales que requiere el equipo de desarrollo puede contrastar con el menor enfoque documentario de las metodologías ágiles (Jarzębowicz, A., & Weichbroth, P., 2021).

Tiempo de implementación. Para una empresa como MS4M con 10 años de experiencia desarrollando software, a la vez con una cultura marcada por la orientación al

resultado, es conveniente que la implementación pueda dividirse en etapas de planificación que puede durar un mes, opcionalmente optar por un piloto que puede demorar de dos a tres meses, la expansión a otros equipos del área de desarrollo de tres a seis meses más (Kim, et al., 2021), por lo que desde el pilotaje se puede obtener mayor impacto deseado en la calidad del producto final, en suma, en 9 meses aproximadamente.

6.1.4. Subcontratar el desarrollo de software

Descripción. Una empresa como MS4M que desarrolla software y hardware para un nicho de mercado, puede explorar una solución a sus problemas de calidad de producto en base a la subcontratación de su actividad principal como lo es el desarrollo de software. Las empresas tecnológicas como MS4M podrían tercerizar esta actividad precisamente por la falta de recursos humanos o financieros, en el primero el problema puede ser de calidad o de cantidad; o puede subcontratarlo por la falta de habilidades especializadas de su equipo de desarrollo especialmente en tecnologías emergentes o experiencia en un área de conocimiento específica, o simplemente para obtener mayor eficiencia concentrándose en la calidad del producto final (Govil, et al., 2023). La subcontratación puede ser total, que abarque a todo el equipo de desarrolladores incluyendo las posiciones de supervisión, dirección y soporte, o parcialmente manteniendo los niveles de superiores. Inclusive puede ser externa al país buscando mejores precios o eficiencias, o interna con una empresa especializada. Estas diferentes opciones permiten de que las empresas en general puedan afrontar problemas coyunturales o de fondo, cada uno con diferente impacto y riesgos asociados.

Beneficios esperados. Las empresas de tecnología recurren a la subcontratación de sus actividades principales mayoritariamente buscando una mayor rentabilidad económica, búsqueda de recursos humanos más expertos o para obtener un mayor tiempo de desarrollo,

esto último significa que cuando su equipo está durmiendo, una empresa extranjera continua el trabajo en diferente horario. Pocos buscan una mayor calidad del producto y, de hecho, es un problema que deben afrontar. (Rahman, et al., 2021) pero, sin embargo, puede permitir a MS4M un mayor enfoque liberando y proporcionando mayor cantidad de tiempo a la gestión y a las actividades de innovación en la forma en que se encara al mercado con nuevos productos o soluciones.

Desafíos. Son muchos los desafíos en este contexto, uno de ellos tiene carácter laboral, ya que en Perú no se podía subcontratar totalmente las actividades que pertenecen al giro de la empresa con diversas modificaciones y actualizaciones (Perú C. d., 2021). Sin embargo las estrategias conllevan a tomar una parte de cada proyecto para que pueda ser realizado parcialmente por parte de MS4M y complementada por otra mayormente especializada, sea local o extranjera, y a partir se derivan otros desafíos relacionados a la comunicación, coordinación y sincronización sobre los requisitos funcionales que ocasionan exceso de tiempos y fallas de calidad; adicionalmente la gestión del conocimiento relacionado a la documentación y sobre todo, a la propiedad intelectual y condiciones de confidencialidad relacionadas a las patentes.

Todo esto conlleva a problemas que se tienen que identificar y gestionar tanto con una mayor complejidad en la selección adecuada y confiable, como en los recursos a asignarse para ello (Rahman, et al., 2021) situación que puede ser agravada al considerar la subcontratación fuera del país, donde además se acrecientan los problemas relacionados a las diferencias culturales donde la complejidad de las comunicaciones, entendimiento de requerimientos funcionales y no funcionales y, riesgos asociados a la gestión de la propiedad intelectual son mayores (Gupta, et al., 2022).

Esto último representa un riesgo mayor que MS4M debe asumir como la fuga del conocimiento intelectual puede darse de forma intencional o no intencional, explícita o implícita, o bajos diferentes niveles de complejidad a ser dimensionada y gestionada; esto muy a pesar de estrategias que MS4M puede adoptar tal como una adecuada gestión de la propiedad intelectual, selección cuidadosa de los socios a incorporar, acuerdos de confidencialidad y establecer una cultura de protección del conocimiento, sin embargo a pesar de todo ello, el riesgo aún persiste (Edris et al., 2024). Inclusive, existen estrategias para la protección de la propiedad intelectual en las cuales las empresas que colaboran en los proyectos sólo lo hacen bajo una forma parcial muy difícil de integrar hacia el producto final con accesos controlados a la parte complementaria del conocimiento compartido, y a pesar de ello, el riesgo persiste (Kashevarova, 2024).

6.1.5. Incorporar desarrolladores de mayor experiencia

Descripción. En desarrollo de software, el nivel de experiencia no es suficiente, sino más bien una combinación de competencias, certificaciones y experiencia que se traducen en rapidez para solucionar diferentes tipos de problemas, con alta calidad y menor impacto (Baloch, et al., 2021). Sin embargo, la realidad de MS4M respecto a sus cuadros en el área de investigación y desarrollo de productos se muestra en la siguiente tabla 30.

Tabla 30*Experiencia y certificación de personal de investigación y desarrollo*

<i>Certificaciones / Experiencia</i>	<i>Menos de un año</i>	<i>Entre 1 y dos años</i>	<i>Entre dos y cuatro años</i>	<i>Entre cuatro y cinco años</i>	<i>Mas de cinco años</i>	<i>Total</i>
Maestría	0	0	5	0	5	10
Más de seis certificaciones	0	0	3	1	6	10
Entre cuatro y seis certificaciones	0	3	17	0	4	24
Entre dos y tres certificaciones	3	2	0	0	1	6
Con una certificación	0	0	0	0	0	0
Sin ninguna certificación	1	0	0	0	0	1
Total	4	5	20	1	11	51

Nota. Elaborado desde el área de Recursos Humanos MS4M. Empresa MS4M. (2023), (No publicado)

MS4M tiene un grupo de experiencia en su desarrollo de software que representa el 20% del total con grado de maestría y cerca de 50% con certificaciones de especialización, por lo que las capacidades técnicas de los desarrolladores tienen poco espacio para aportes externos adicionales.

MS4M también tiene su equipo de desarrollo de software ubicado en la región de Cajamarca, desde donde más del 90% proviene de la Universidad Nacional de Cajamarca. Allí ha forjado una cultura de superación y proyección para recién egresados captándolos e incorporándolos a una carrera de evolución y retos permanente, desde donde han podido llegar hasta niveles directivos con un aprendizaje permanente.

Las estrategias en las que se relaciona al personal de mayor experiencia en los equipos de desarrollo de software los involucran al establecimiento de una cultura que

permita altos niveles de comunicación, liderazgo en reuniones de trabajo, inteligencia emocional para afrontar situaciones difíciles y sobre cargas de trabajo coyunturales (IT, 2023) situaciones que por lo general los desarrolladores menos experimentados abordan de una manera menos eficiente, sin que esto signifique que no cuentan con las mismas competencias técnicas que los más experimentados, quienes aportan además de lo descrito, un mayor dominio de las aplicaciones y control, pero sólo pueden verse como adelantados en una carrera donde los menos expertos partieron después, ya que los más expertos aún permanecen capacitándose y aprendiendo nuevas arquitecturas, conceptos, aplicaciones de integración y plataformas (Hicks, et al., 2024).

Beneficios esperados. El aporte de desarrolladores más experimentados en MS4M permitiría una mayor adaptación y aportes de diferentes puntos de vista para la resolución de problemas, una productividad diferente en la atención de tickets, pero, sobre todo, un mejor manejo emocional para afrontar situaciones difíciles construyendo una cultura más eficiente en la que los menos experimentados puedan emular (Hicks, et al., 2024).

Desafíos. Dentro de los desafíos más importantes son el financiero, lo que incorporar desarrolladores podría desequilibrar las políticas salariales y la línea de carrera, ya que se debe de competir en el mercado por estos perfiles, donde posibles disconformidades con la cultura creada en MS4M de superación y líneas de carrera claras para la evolución de los proyectos individuales de cada desarrollador. Una cultura orientada al esfuerzo y a la superación fuerte puede debilitarse si los niveles de productividad son constantes (Rahman, et al., 2021), sobre todo con el aporte de externos el mensaje es diferente.

6.1.6. Refactorizar el código de los productos

Descripción. La refactorización de código es el proceso de mejorar la estructura interna de las líneas de programación sin alterar la funcionalidad ni los requisitos que satisface, con el objetivo de que sea más fácil de entender, modificar, integrar, modular y escalar, mejorando su gestión integral. Con la eliminación de la deuda técnica que acarrea en base a anti-patrones y errores de código, reduce la complejidad y mejora la calidad de código, permitiendo una menor necesidad de soporte y una mejor evolución en el tiempo (Smiri, et al., 2022).

Contrariamente a lo que tradicionalmente se ha entendido por la refactorización de código, no sólo es aplicable a los productos existentes. Esta técnica puede aplicarse como una práctica permanente y estandarizada durante los procesos de creación de nuevos productos, con intervalos tempranos de revisión permanente para disminuir la deuda técnica y permitir que el producto final sea el resultado de muchas iteraciones de refactorización de código con un mínimo de defectos de calidad (Morales et al., 2017).

Beneficios esperados. Refactorizar el código existente, e implementar prácticas de refactorización en intervalos tempranos de desarrollo de software permite que los costos de mantenimiento y soporte se vean reducidos en el largo plazo, con una inversión de corto plazo que consiste en mayores tiempos de desarrollo producto de revisiones y refactorizaciones (Iammarino, et al., 2021). Se espera que los productos refactorizados tengan un menor costo de soporte y mantención que lo que no estén refactorizados, con un mínimo de deuda técnica.

Desafíos. Un desafío bastante importante es la definición sobre lo que hay que refactorizar. Actualmente se encuentran muchas herramientas automatizadas para ejecutar las

labores de refactorización y cada una entiende criterios desde sintaxis hasta “*code smells*”, sin embargo, la calidad de código puede ser buena pero la arquitectura que la sostiene no. En ecosistemas donde el producto final es parte de otro grupo de soluciones la refactorización puede quedar simplemente insuficiente para solucionar la complejidad de una arquitectura subyacente. Otros simplemente pueden desprenderse de depender mucho de un equipo muy calificado para no introducir nuevos errores en los procesos de refactorización, un patrocinador que pueda justificar ante el cliente nuevas necesidades de refactorización y un abuso de este proceso si no se aplican reglas claras para una planificación y ejecución eficiente, sobre todo en entornos ágiles, donde se pueden aceptar nuevos requisitos, la refactorización puede resultar improductiva y frustrante (Smiri, et al., 2022).

Tabla 31

Comparación de alternativas de solución (parte 1)

<i>Alternativa / Criterio</i>	<i>Beneficios</i>	<i>Desafíos</i>	<i>Tiempo de implementación</i>
A Implementar un modelo de gobernanza organizacional	Al enfocarse en la calidad del producto final, considerando la supervisión de equipo (retroalimentación), la rendición de cuentas (desarrollo y aprendizaje), control y discusiones de lecciones aprendidas, alineamiento a los objetivos de la organización y finalmente procesos guías de mitigación de riesgos, permitirá a MS4M establecer aspectos de gobernanza críticos fortaleciendo el liderazgo, gestión del equipo y mejora en la rendición de cuentas, retroalimentación y comunicación interna.	Actualmente MS4M cuenta con un 80% de sus procesos en nivel iniciales e incompletos, - lo que implica un riesgo alto en la adopción de criterios de gobernanza.	

Nota. Elaboración propia

Tabla 32*Comparación de alternativas de solución (parte 2)*

<i>Alternativa / Criterio</i>	<i>Beneficios</i>	<i>Desafíos</i>	<i>Tiempo de implementación</i>
B Implementar un modelo de arquitectura empresarial	La implementación del modelo de arquitectura empresarial en MS4M permitirá formalizar procesos, así como guías y métodos de trabajo específico, aportando en ámbitos de desarrollo como soluciones a problemas a largo plazo e integraciones de soluciones.	MS4M deberá medir los niveles de calidad, considerar el tiempo de implementación y su alcance, con el objetivo de medir de su impacto. Asimismo, deberá evitar la creación de expectativas sobrevaloradas.	Mayor
C Implementar metodologías ágiles	La implementación de metodologías en MS4M, mejoraría la gestión de la deuda técnica a través del desarrollo iterativos, colaboración, comunicación y mejora continua en el desarrollo de software. Identificando de manera temprana las brechas en el desarrollo, favoreciendo la calidad del producto final y reduciendo el mantenimiento.	MS4M deberá evaluar los requisitos funcionales y no funcionales, los cuales deberán ser detallados y priorizados, así como desafíos culturales (considerando mayor atención en la gestión de cambio)	Medio
D Subcontratar el desarrollo de software	De subcontratar el desarrollo MS4M incrementaría su rentabilidad a través de recursos expertos, asimismo permitirá aprovechar tiempos de desarrollo (horarios internacionales).	De aplicar la subcontratación MS4M deberá considerar que el desarrollo del producto será de manera parcial y complementado por otra parte especializada, lo cual podría involucrar desafíos de comunicación, coordinación entre otro, aumentando la complejidad de gestión. Así como exposición a fuga de conocimiento intelectual.	-
E Incorporar desarrolladores de mayor experiencia	De incorporar desarrolladores con mayor experiencia en MS4M, permitirá obtener aportes en puntos de vista aportando en la productividad de atención tiques, favoreciendo la construcción de una cultura eficiente que pueda emularse por el personal menos experimentado.	De contratar personal experimentado podría desequilibrar las políticas salariales, que puede afectar a la cultura actual de manera negativa.	-
F Refactorizar el código de los productos	Al implementar la refactorización de código, en los productos existentes y a los nuevos productos permitirá reducir defectos en la calidad de estos, reduciendo costos de mantenimiento y soporte, aportando en la reducción de la deuda técnica.	Esta implementación podría ser insuficiente para solucionar la complejidad de los productos, además que esto requeriría personal altamente calificado para evitar introducir nuevos errores.	-

Nota. Elaboración propia.

6.2. Evaluación de alternativas

Considerando las alternativas de solución identificadas, el equipo a través del análisis realizado en el capítulo V, y se consideró analizar el nivel de impacto en los tópicos evaluados por el personal de soporte técnico de MS4M.

Los tópicos evaluados fueron: Calidad de código, complejidad y/o duplicidad de código, documentación, arquitectura, tecnología obsoleta, control de calidad y despliegue.

6.2.1. Modelo de Evaluación

El modelo de evaluación aplicada en la presente investigación será el Proceso de Jerarquía Analítica – AHP, el cual permite realizar comparaciones de pares entre un conjunto de alternativas en cada criterio analizado (Domínguez, et al., 2021). Estas comparaciones se realizarán en una escala de juicios, que tendrán los siguientes valores:

Tabla 33

Escala de Juicios

<i>Valor</i>	<i>Escala Verbal</i>	<i>Descripción</i>
1	Igualmente, Importante	Par de elementos contribuyen en igual medida al objetivo
3	Moderadamente Importante	Preferencia leve de un elemento sobre otro
5	Fuertemente Importante	preferencia fuerte de un elemento sobre el otro
7	Importancia muy fuerte o demostrada	Mucha más preferencia de un elemento sobre otro. Predominancia demostrada
9	Importancia Extremadamente Fuerte	Preferencia clara y absoluta de un elemento sobre el otro
2,4,6,8	Valores intermedios entre dos juicios	

Nota. Tomado del Artículo “Aplicación de los métodos AMEF-TOPSIS-AHP para determinar el RPN” Mundo FESC (2021)

6.2.2. Criterios de Evaluación

Para realizar la evaluación de los criterios identificados, se ha considerado el nivel de impacto (alineado a la escala de juicios de la metodología AHP) sobre dichos criterios, los cuales serán evaluados en las siguientes tablas, siendo:

Impacto en la calidad de código.

Tabla 34

Comparación de criterios de Impacto de calidad de código (Ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	0,50	0,33	3,00	0,14	0,20
<i>Implementar un modelo de arquitectura empresarial</i>	2,00	1,00	0,14	3,00	0,20	0,20
<i>Implementar metodologías ágiles</i>	3,00	7,00	1,00	7,00	0,50	0,33
<i>Subcontratar el desarrollo de software</i>	0,33	0,33	0,14	1,00	0,20	0,17
<i>Incorporar desarrolladores de mayor experiencia</i>	7,00	5,00	2,00	5,00	1,00	0,33
<i>Refactorizar el código de los productos</i>	5,00	5,00	3,00	6,00	3,00	1,00
Σ Columnas	18,33	18,83	6,62	25,00	5,04	2,23

Nota. Elaboración propia.

En la tabla 34, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en la calidad del código. Cada alternativa se

compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 35

Normalización de criterio de Impacto de calidad de código

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,03	0,05	0,12	0,03	0,09	0,06
<i>Implementar un modelo de arquitectura empresarial</i>	0,11	0,05	0,02	0,12	0,04	0,09	0,07
<i>Implementar metodologías ágiles</i>	0,16	0,37	0,15	0,28	0,10	0,15	0,20
<i>Subcontratar el desarrollo de software</i>	0,02	0,02	0,02	0,04	0,04	0,07	0,04
<i>Incorporar desarrolladores de mayor experiencia</i>	0,38	0,27	0,30	0,20	0,20	0,15	0,25
<i>Refactorizar el código de los productos</i>	0,27	0,27	0,45	0,24	0,59	0,45	0,38

Nota. Elaboración propia.

En la Tabla 35 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en la calidad del código. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la Tabla 33 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en la calidad del código.

Impacto en la complejidad y/o duplicidad.

Tabla 36

Comparación de criterios: Impacto de complejidad y/o duplicidad (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	0,33	0,14	3,00	0,20	0,20
<i>Implementar un modelo de arquitectura empresarial</i>	3,00	1,00	0,20	3,00	0,20	0,20
<i>Implementar metodologías ágiles</i>	7,00	5,00	1,00	7,00	0,50	0,33
<i>Subcontratar el desarrollo de software</i>	0,33	0,33	0,14	1,00	0,20	0,17
<i>Incorporar desarrolladores de mayor experiencia</i>	5,00	5,00	2,00	5,00	1,00	0,33
<i>Refactorizar el código de los productos</i>	5,00	5,00	3,00	6,00	3,00	1,00
Σ Columnas	21,33	16,67	6,49	25,00	5,10	2,23

Nota. Elaboración propia.

En la tabla 36, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en la complejidad y/o duplicidad. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 37*Normalización de criterio de Impacto de la complejidad y/o duplicidad*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,02	0,02	0,12	0,04	0,09	0,06
<i>Implementar un modelo de arquitectura empresarial</i>	0,16	0,05	0,03	0,12	0,04	0,09	0,08
<i>Implementar metodologías ágiles</i>	0,38	0,27	0,15	0,28	0,10	0,15	0,22
<i>Subcontratar el desarrollo de software</i>	0,02	0,02	0,02	0,04	0,04	0,07	0,04
<i>Incorporar desarrolladores de mayor experiencia</i>	0,27	0,27	0,30	0,20	0,20	0,15	0,23
<i>Refactorizar el código de los productos</i>	0,27	0,27	0,45	0,24	0,59	0,45	0,38

Nota. Elaboración propia.

En la tabla 37 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en la complejidad y/o duplicidad. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 36 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en la complejidad y/o duplicidad.

Impacto en la documentación.

Tabla 38

Comparación de criterios de Impacto en la documentación (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	3,00	2,00	4,00	2,00	2,00
<i>Implementar un modelo de arquitectura empresarial</i>	0,33	1,00	0,33	3,00	3,00	2,00
<i>Implementar metodologías ágiles</i>	0,50	3,00	1,00	4,00	4,00	4,00
<i>Subcontratar el desarrollo de software</i>	0,25	0,33	0,25	1,00	0,33	0,33
<i>Incorporar desarrolladores de mayor experiencia</i>	0,50	0,33	0,25	3,00	1,00	3,00
<i>Refactorizar el código de los productos</i>	0,50	0,50	0,25	3,00	0,33	1,00
Σ Columnas	3,08	8,17	4,08	18,00	10,67	12,33

Nota. Elaboración propia.

En la tabla 38, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en la documentación. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 39*Normalización de criterio de Impacto de la documentación*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,16	0,30	0,16	0,40	0,90	0,33
<i>Implementar un modelo de arquitectura empresarial</i>	0,02	0,05	0,05	0,12	0,59	0,90	0,29
<i>Implementar metodologías ágiles</i>	0,03	0,16	0,15	0,16	0,79	1,79	0,51
<i>Subcontratar el desarrollo de software</i>	0,01	0,02	0,04	0,04	0,07	0,15	0,05
<i>Incorporar desarrolladores de mayor experiencia</i>	0,03	0,02	0,04	0,12	0,20	1,34	0,29
<i>Refactorizar el código de los productos</i>	0,03	0,03	0,04	0,12	0,07	0,45	0,12

Nota. Elaboración propia.

En la tabla 39 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en la documentación. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 38 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en la documentación.

Impacto en la arquitectura.

Tabla 40

Comparación de criterios de Impacto en la arquitectura (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	0,25	0,33	5,00	3,00	5,00
<i>Implementar un modelo de arquitectura empresarial</i>	4,00	1,00	3,00	7,00	3,00	4,00
<i>Implementar metodologías ágiles</i>	3,00	0,33	1,00	5,00	5,00	5,00
<i>Subcontratar el desarrollo de software</i>	0,20	0,14	0,20	1,00	0,33	0,33
<i>Incorporar desarrolladores de mayor experiencia</i>	0,33	0,33	0,20	3,00	1,00	3,00
<i>Refactorizar el código de los productos</i>	0,20	0,25	0,20	3,00	0,33	1,00
Σ Columnas	8,73	2,31	4,93	24,00	12,67	18,33

Nota. Elaboración propia.

En la tabla 40, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en la arquitectura. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 41*Normalización de criterio de Impacto en la arquitectura*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,01	0,05	0,20	0,59	2,24	0,53
<i>Implementar un modelo de arquitectura empresarial</i>	0,22	0,05	0,45	0,28	0,59	1,79	0,57
<i>Implementar metodologías ágiles</i>	0,16	0,02	0,15	0,20	0,99	2,24	0,63
<i>Subcontratar el desarrollo de software</i>	0,01	0,01	0,03	0,04	0,07	0,15	0,05
<i>Incorporar desarrolladores de mayor experiencia</i>	0,02	0,02	0,03	0,12	0,20	1,34	0,29
<i>Refactorizar el código de los productos</i>	0,01	0,01	0,03	0,12	0,07	0,45	0,11

Nota. Elaboración propia.

En la tabla 41 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en la arquitectura. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 40 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en la arquitectura.

Impacto en la tecnología obsoleta.

Tabla 42

Comparación de criterios de Impacto en la tecnología obsoleta (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	0,33	0,33	3,00	0,20	2,00
<i>Implementar un modelo de arquitectura empresarial</i>	3,00	1,00	3,00	3,00	4,00	3,00
<i>Implementar metodologías ágiles</i>	3,00	0,33	1,00	3,00	0,33	3,00
<i>Subcontratar el desarrollo de software</i>	0,33	0,33	0,33	1,00	0,20	0,50
<i>Incorporar desarrolladores de mayor experiencia</i>	5,00	0,25	3,00	5,00	1,00	5,00
<i>Refactorizar el código de los productos</i>	0,50	0,33	0,33	2,00	0,20	1,00
Σ Columnas	12,83	2,58	8,00	17,00	5,93	14,50

Nota. Elaboración propia.

En la tabla 42, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en la tecnología obsoleta. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 43*Normalización de criterio de Impacto en la tecnología obsoleta*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,02	0,05	0,12	0,04	0,90	0,20
<i>Implementar un modelo de arquitectura empresarial</i>	0,16	0,05	0,45	0,12	0,79	1,34	0,49
<i>Implementar metodologías ágiles</i>	0,16	0,02	0,15	0,12	0,07	1,34	0,31
<i>Subcontratar el desarrollo de software</i>	0,02	0,02	0,05	0,04	0,04	0,22	0,06
<i>Incorporar desarrolladores de mayor experiencia</i>	0,27	0,01	0,45	0,20	0,20	2,24	0,56
<i>Refactorizar el código de los productos</i>	0,03	0,02	0,05	0,08	0,04	0,45	0,11

Nota. Elaboración propia.

En la tabla 43 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en la tecnología obsoleta. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 42 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en la tecnología obsoleta.

Impacto en el control de calidad.

Tabla 44

Comparación de criterios de Impacto en el control de calidad (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	3,00	3,00	3,00	3,00	3,00
<i>Implementar un modelo de arquitectura empresarial</i>	0,33	1,00	3,00	3,00	0,33	2,00
<i>Implementar metodologías ágiles</i>	0,33	0,33	1,00	2,00	0,33	0,50
<i>Subcontratar el desarrollo de software</i>	0,33	0,33	0,50	1,00	0,33	0,33
<i>Incorporar desarrolladores de mayor experiencia</i>	0,33	3,00	3,00	3,00	1,00	3,00
<i>Refactorizar el código de los productos</i>	0,33	0,50	2,00	3,00	0,33	1,00
Σ Columnas	2,67	8,17	12,50	15,00	5,33	9,83

Nota. Elaboración propia.

En la tabla 44, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en el control de calidad. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 45*Normalización de criterio de Impacto en el control de calidad*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,16	0,45	0,12	0,59	1,34	0,45
<i>Implementar un modelo de arquitectura empresarial</i>	0,02	0,05	0,45	0,12	0,07	0,90	0,27
<i>Implementar metodologías ágiles</i>	0,02	0,02	0,15	0,08	0,07	0,22	0,09
<i>Subcontratar el desarrollo de software</i>	0,02	0,02	0,08	0,04	0,07	0,15	0,06
<i>Incorporar desarrolladores de mayor experiencia</i>	0,02	0,16	0,45	0,12	0,20	1,34	0,38
<i>Refactorizar el código de los productos</i>	0,02	0,03	0,30	0,12	0,07	0,45	0,16

Nota. Elaboración propia.

En la tabla 45 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en el control de calidad. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 44 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en el control de calidad.

Impacto en el despliegue.

Tabla 46

Comparación de criterios de Impacto en el despliegue (ponderado)

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>
<i>Implementar un modelo de gobernanza organizacional</i>	1,00	3,00	3,00	3,00	3,00	3,00
<i>Implementar un modelo de arquitectura empresarial</i>	0,33	1,00	3,00	3,00	3,00	5,00
<i>Implementar metodologías ágiles</i>	0,33	0,33	1,00	5,00	5,00	5,00
<i>Subcontratar el desarrollo de software</i>	0,33	0,33	0,20	1,00	0,33	0,33
<i>Incorporar desarrolladores de mayor experiencia</i>	0,33	0,33	0,20	3,00	1,00	3,00
<i>Refactorizar el código de los productos</i>	0,33	0,20	0,20	3,00	0,33	1,00
Σ Columnas	2,67	5,20	7,60	18,00	12,67	17,33

Nota. Elaboración propia.

En la tabla 46, se presenta la matriz de comparación de pares de las alternativas de solución con respecto al criterio de Impacto en el despliegue. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, según el criterio mencionado, utilizando la escala de importancia de la tabla 33.

Tabla 47*Normalización de criterio de Impacto en el despliegue*

	<i>Implementar un modelo de gobernanza organizacional</i>	<i>Implementar un modelo de arquitectura empresarial</i>	<i>Implementar metodologías ágiles</i>	<i>Subcontratar el desarrollo de software</i>	<i>Incorporar desarrolladores de mayor experiencia</i>	<i>Refactorizar el código de los productos</i>	<i>Vector Prom.</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,05	0,16	0,45	0,12	0,59	1,34	0,45
<i>Implementar un modelo de arquitectura empresarial</i>	0,02	0,05	0,45	0,12	0,59	2,24	0,58
<i>Implementar metodologías ágiles</i>	0,02	0,02	0,15	0,20	0,99	2,24	0,60
<i>Subcontratar el desarrollo de software</i>	0,02	0,02	0,03	0,04	0,07	0,15	0,05
<i>Incorporar desarrolladores de mayor experiencia</i>	0,02	0,02	0,03	0,12	0,20	1,34	0,29
<i>Refactorizar el código de los productos</i>	0,02	0,01	0,03	0,12	0,07	0,45	0,12

Nota. Elaboración propia.

En la tabla 47 se muestra la normalización de la matriz de pares de las alternativas de solución con respecto al criterio de Impacto en el despliegue. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 46 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis de las alternativas en términos de su impacto en el despliegue.

Tabla 48*Comparación de criterios Técnicos (ponderado)*

	<i>Impacto en la calidad de código</i>	<i>Impacto en la complejidad y/o duplicidad</i>	<i>Impacto en la documentación</i>	<i>Impacto en la arquitectura</i>	<i>Impacto en la tecnología obsoleta</i>	<i>Impacto en el control de calidad</i>	<i>Impacto en el despliegue</i>
<i>Impacto en la calidad de código</i>	1,00	3,00	5,00	0,20	0,50	2,00	3,00
<i>Impacto en la complejidad y/o duplicidad</i>	0,33	1,00	2,00	0,20	0,33	0,20	0,50
<i>Impacto en la documentación</i>	0,20	0,50	1,00	0,20	0,33	0,20	2,00
<i>Impacto en la arquitectura</i>	5,00	5,00	5,00	1,00	3,00	4,00	4,00
<i>Impacto en la tecnología obsoleta</i>	2,00	3,00	3,00	0,33	1,00	0,50	3,00
<i>Impacto en el control de calidad</i>	0,50	5,00	5,00	0,25	2,00	1,00	3,00
<i>Impacto en el despliegue</i>	0,33	2,00	0,50	0,25	0,33	0,33	1,00
Σ Columnas	9,37	19,50	21,50	2,43	7,50	8,23	16,50

Nota. Elaboración propia.

En la tabla 48, se presenta la matriz de comparación de pares entre los criterios evaluados. Cada alternativa se compara con las demás, teniendo en cuenta su importancia en pares, utilizando la escala de importancia de la tabla 33.

Tabla 49*Normalización de criterios Técnicos*

	<i>Impacto en la calidad de código</i>	<i>Impacto en la complejidad y/o duplicidad</i>	<i>Impacto en la documentación</i>	<i>Impacto en la arquitectura</i>	<i>Impacto en la tecnología obsoleta</i>	<i>Impacto en el control de calidad</i>	<i>Impacto en el despliegue</i>	<i>Vector Prom.</i>
<i>Impacto en la calidad de código</i>	0,05	0,16	0,76	0,01	0,10	0,90	0,18	0,31
<i>Impacto en la complejidad y/o duplicidad</i>	0,02	0,05	0,30	0,01	0,07	0,09	0,03	0,08
<i>Impacto en la documentación</i>	0,01	0,03	0,15	0,01	0,07	0,09	0,12	0,07
<i>Impacto en la arquitectura</i>	0,27	0,27	0,76	0,04	0,59	1,79	0,24	0,57
<i>Impacto en la tecnología obsoleta</i>	0,11	0,16	0,45	0,01	0,20	0,22	0,18	0,19
<i>Impacto en el control de calidad</i>	0,03	0,27	0,76	0,01	0,40	0,45	0,18	0,30
<i>Impacto en el despliegue</i>	0,02	0,11	0,08	0,01	0,07	0,15	0,06	0,07

Nota. Elaboración propia.

En la tabla 49 se muestra la normalización de la matriz de pares entre los criterios evaluados. Cada valor en la matriz ha sido ajustado para reflejar su importancia en relación con los otros, se obtiene dividiendo cada elemento de la matriz de pares de la tabla 48 entre la suma de los elementos de su respectiva columna. Esta normalización permite la comparación y análisis entre los criterios evaluados.

Tabla 50*Matriz de Priorización Criterios Técnicos*

	<i>Impacto en la calidad de código</i>	<i>Impacto en la complejidad y/o duplicidad</i>	<i>Impacto en la documentación</i>	<i>Impacto en la arquitectura</i>	<i>Impacto en la tecnología obsoleta</i>	<i>Impacto en el control de calidad</i>	<i>Impacto en el despliegue</i>	<i>Total</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,06	0,06	0,33	0,53	0,20	0,45	0,45	0,65
<i>Implementar un modelo de arquitectura empresarial</i>	0,07	0,08	0,29	0,57	0,49	0,27	0,58	0,71
<i>Implementar metodologías ágiles</i>	0,20	0,22	0,51	0,63	0,31	0,09	0,60	0,74
<i>Subcontratar el desarrollo de software</i>	0,04	0,04	0,05	0,05	0,06	0,06	0,05	0,09
<i>Incorporar desarrolladores de mayor experiencia</i>	0,25	0,23	0,29	0,29	0,56	0,38	0,29	0,59
<i>Refactorizar el código de los productos</i>	0,38	0,38	0,12	0,11	0,11	0,16	0,12	0,32
<i>Ponderación</i>	0,31	0,08	0,07	0,57	0,19	0,30	0,30	

Nota. Elaboración propia.

En la tabla 50 se muestra la ponderación resultante de cada uno de los criterios por la ponderación de las alternativas, mostrando el puntaje respectivo por cada alternativa.

6.2.3. Resultados de Evaluación

A través del análisis de valoración basado en la Metodología de Ponderación Jerárquica (AHP), se evaluaron las diferentes alternativas de solución en función de los criterios técnicos, económicos y comerciales descritos. Los resultados ponderados son los siguientes:

Implementar metodologías ágiles (0.74). Esta alternativa de solución en MS4M, permitirá incentivar la colaboración, flexibilidad y la adaptabilidad, aportando en la calidad del código, así como la simplificación de los mecanismos de integración en los entornos de desarrollo y priorizando la documentación necesaria. Lo cual contribuirá en la reducción. Estas prácticas también pueden ayudar a reducir la complejidad y duplicidad de código, así como a acelerar el proceso de despliegue. No obstante, se debe realizar seguimiento a los procesos de control de calidad de productos, a fin de evitar saturación innecesaria de los recursos, lo cual implicaría impacto en el presupuesto de los proyectos.

Implementar un modelo de gobernanza organizacional (0.65). Esta implementación favorecerá la gestión y coordinación entre los equipos y áreas involucradas en el desarrollo de productos, contribuyendo a la calidad de código, la reducción de complejidad y duplicidad. Sin embargo, esta solución presenta un impacto menor en la arquitectura y la tecnología actual en MS4M, considerando que esta alternativa no ejecutaría cambios inmediatos en los aspectos mencionados. Finalmente, se debe considerar que esta implementación tendría un costo considerable en relación a la adopción de metodologías ágiles considerando tiempo y costo en MS4M.

Implementar un modelo de arquitectura empresarial (0.71). Mediante la adopción de un marco arquitectónico bien definido en MS4M ayudará a mejorar la calidad del código, la arquitectura y la documentación, de igual manera reducirá la complejidad y duplicidad de código. Adicionalmente, contribuirá a que la organización pueda eliminar la tecnología obsoleta, y potenciar el control de calidad. Sin embargo, implementarla puede requerir un gran esfuerzo inicial y un tiempo prolongado. Por otro lado, busca optimizar internamente la infraestructura del negocio.

Tabla 51*Resultados de evaluación*

	<i>Impacto en la calidad de código</i>	<i>Impacto en la complejidad y/o duplicidad</i>	<i>Impacto en la documentación</i>	<i>Impacto en la arquitectura</i>	<i>Impacto en la tecnología obsoleta</i>	<i>Impacto en el control de calidad</i>	<i>Impacto en el despliegue</i>	<i>Total</i>
<i>Implementar un modelo de gobernanza organizacional</i>	0,06	0,06	0,33	0,53	0,20	0,45	0,45	0,65
<i>Implementar un modelo de arquitectura empresarial</i>	0,07	0,08	0,29	0,57	0,49	0,27	0,58	0,71
<i>Implementar metodologías ágiles</i>	0,20	0,22	0,51	0,63	0,31	0,09	0,60	0,74
<i>Subcontratar el desarrollo de software</i>	0,04	0,04	0,05	0,05	0,06	0,06	0,05	0,09
<i>Incorporar desarrolladores de mayor experiencia</i>	0,25	0,23	0,29	0,29	0,56	0,38	0,29	0,59
<i>Refactorizar el código de los productos</i>	0,38	0,38	0,12	0,11	0,11	0,16	0,12	0,32
<i>Ponderación</i>	0,31	0,08	0,07	0,57	0,19	0,30	0,30	

Nota. Elaboración propia.

6.2.4. Decisión Final

Tras la aplicación de la metodología de ponderación jerárquica AHP y la evaluación detallada de los criterios técnicos, económicos y comerciales, así como las alternativas de solución propuestas para MS4M dedicada al desarrollo de software y hardware en el entorno minero, se ha llegado a la decisión final y los resultados ponderados son los siguientes: (1) Implementar metodologías ágiles: 0.74, (2) Implementar un modelo de arquitectura empresarial: 0.71, (3) Implementar un modelo de gobernanza organizacional:0.65.

Sustento de la Decisión.

Respecto a los criterios técnicos: Las metodologías ágiles obtuvieron la puntuación más alta en estos aspectos, lo que indica su capacidad para mejorar la eficiencia y efectividad en el desarrollo de software y hardware en un entorno minero.

Respecto a los criterios económicos: Las metodologías ágiles también destacaron en estos criterios al ofrecer una mayor agilidad y rapidez en la implementación, lo que puede resultar en una optimización de recursos y tiempos.

Respecto a los criterios comerciales: Las metodologías ágiles demostraron ser beneficiosas al adaptarse rápidamente a las necesidades del mercado y los clientes, lo que puede impulsar la competitividad y satisfacción del cliente.

Asimismo, y dada la puntuación de las metodologías ágiles, esta se posiciona como la alternativa de solución priorizada. Su capacidad para mejorar la calidad del código, reducir la complejidad, acelerar el despliegue, adaptarse a los cambios del mercado y clientes, así como optimizar costos y tiempos de implementación, la establece como alternativa inicial que permitirá a MS4M implementar marcos de referencia.

En contraste, aunque tanto el modelo de gobernanza organizacional como el modelo de arquitectura empresarial presentan ventajas específicas, no logran igualar el impacto integral que las metodologías ágiles ofrecen en términos técnicos, económicos y comerciales. Sin embargo es necesario precisar que la diferencia en su ponderación final puede representar una oportunidad de implementar opciones combinadas o integradas de estos tres elementos de forma tal de complementar un marco de referencia con un modelo de arquitectura empresarial orientada al desarrollo de software y lineamientos de gobierno que soporten las nuevas prácticas a introducirse.

Capítulo VII: Solución Propuesta

Las alternativas de solución propuestas descritas en el capítulo anterior consideran un enfoque de implementación de metodologías ágiles, aplicaciones políticas de gobierno sobre el desarrollo de software y la arquitectura empresarial que lo sostenga. Se han identificado múltiples desafíos provenientes de los despliegues de marcos de referencia ágiles como DevSecOps provienen de 4 fuentes principales a ser abordadas, las cuales están relacionados a las personas, las prácticas, las herramientas y la infraestructura (Rajapakse, et al., 2022), así mismo, existe la necesidad según se manifiesta, de generar herramientas de control sobre estos 4 elementos que permitan que estos desafíos se puedan superar durante la implementación y vigencia del marco de referencia que afectará a todo el ciclo de vida del desarrollo de software.

La combinación de estos elementos tales como la cultura organizacional, nivel de madurez y el grado de automatización de las organizaciones, deben de ser abordados según se indica desde diferentes perspectivas que corresponden a lineamientos de gobernanza, como estructura jerárquica que soporten las prácticas de desarrollo de software bajo el enfoque de DevSecOps. La solución propuesta abordará estos tres aspectos. Descritos a continuación.

7.1. Lineamientos, políticas base de gobierno sobre el desarrollo de software

Las estrategias y objetivos de los procesos de desarrollo de software deben de ser definidos antes de estandarizar o implementar cualquier marco de referencia, para ello se recomienda los siguientes lineamientos que gobiernen dichos procesos.

Alineamiento a la estrategia corporativa. Todas las iniciativas que comiencen los procesos de desarrollo de productos digitales estarán relacionadas con los objetivos estratégicos de evolución de productos, satisfacción del cliente o de penetración de mercados,

estando definidas en los alcances de cada proyecto, estos a su vez se alinean con los objetivos comerciales de MS4M. Las iniciativas o requerimientos deben de clasificarse según el origen, y estas podrán medirse según:

$$\text{Alineamiento} = 100\% \frac{\text{Número de iniciativas desarrolladas alineadas}}{\text{Número de iniciativas desarrolladas}}$$

Debido a la escasa madurez de los procesos en MS4M, se espera una evolución gradual de 60% en el primer año, 80% en el segundo y superior al 95% después del tercer año.

Enfoque en la innovación como valor corporativo. Debido a que la innovación es un valor de MS4M, todas las atenciones a requerimientos de desarrollo de productos digitales deben de buscar una mejor y nueva forma de solucionar una necesidad del mercado o del cliente mediante la búsqueda constante de prácticas, tecnologías o recursos disponibles en la industria. Todas las iniciativas o requerimientos deben de ser medidos en alto nivel de la siguiente forma:

$$\text{Innovación} = 100\% \frac{\text{Número de iniciativas desarrolladas considerada como innovadora}}{\text{Número de iniciativas desarrolladas}}$$

De la misma forma, se sugiere el escalamiento gradual de 60%, 80 y superior a 90% en los siguientes tres años.

Adaptación a la necesidad del mercado/clientes como valor corporativo. Se debe priorizar la gestión del valor con productos digitales, antes de iniciar el desarrollo es fundamental comprender en profundidad el mercado objetivo, las necesidades del cliente, las tendencias del mercado, así como la competencia. Esto implica realizar un análisis de mercado detallado y definir claramente la propuesta de valor del producto digital a construir. De la misma forma, es necesario identificar y documentar los requisitos de negocio y los

casos de uso del producto digital. Esto incluye entender las funcionalidades principales del producto, flujos de trabajo, requisitos no funcionales y de cumplimiento. Puede medirse en alto nivel desde:

$$\text{Enfoque a mercado} = 100\% \frac{\text{Número de iniciativas desarrolladas a consecuencia de requerimientos de clientes o mercado}}{\text{Número de iniciativas desarrolladas}}$$

Se recomienda iniciar con un 100% de meta debido a la importancia en la calidad y costos asociados.

Garantía de calidad y seguridad, para implementar procesos y estándares de seguridad y calidad en todo el ciclo de vida de del desarrollo de software, como valor corporativo. En MS4M se deben de aplicar principios de diseño centrados en el usuario para garantizar que el producto digital sea intuitivo, fácil de usar y satisfaga las necesidades reales de los usuarios. Esto implicara realizar pruebas de usabilidad y recopilar retroalimentación de los usuarios durante todo el proceso de desarrollo. De igual forma, integrar consideraciones de seguridad y privacidad desde el diseño, asegurando que el producto digital cumpla con los estándares y regulaciones aplicables; esto implicará implementar medidas de seguridad robustas para proteger el código y garantizar la confidencialidad, integridad y disponibilidad de su contenido. Los productos digitales de MS4M deben de garantizar calidad y seguridad a sus clientes.

Una adecuada forma de medición a alto nivel es:

$$\text{Calidad} = 100\% \frac{\text{Número de iniciativas desarrolladas sin errores}}{\text{Número de iniciativas desarrolladas}}$$

$$\text{Seguridad} = 100\% \frac{\text{Número de iniciativas desarrolladas sin brechas}}{\text{Número de iniciativas desarrolladas}}$$

El estado inicial de los productos digitales que desarrolla MS4M permite sugerir un inicio de 80%, 90% y superior a 95% para los siguientes tres años, sin embargo, en temas relacionados a seguridad, se sugiere perseguir el 100% desde el primer año, debido a su potencial de daño en las relaciones con los clientes.

Colaboración y comunicación efectiva, para fomentar una cultura de soporte y trabajo colaborativo, acorde a lo requerido por metodologías ágiles. MS4M debe facilitar medios y canales de comunicación, físicos y virtuales, para promover el intercambio de ideas, puntos de vista y de conocimientos para abordar los problemas, desafíos o proyectos dentro del proceso de desarrollo de productos digitales. Puede medirse a alto nivel de la siguiente forma:

$$\text{Colaboración} = \frac{\text{Número de interacciones en plataforma de colaboración}}{\text{Número de iniciativas desarrolladas de alta complejidad}}$$

Se sugiere un número superior a 100 interacciones en proyectos complejos.

MS4M debe incorporar los riesgos inherentes al desarrollo de productos digitales dentro de la gestión de riesgos corporativa, para ello debe implementar lineamientos base para los siguientes aspectos.

Roles y responsabilidades sobre la gestión de riesgos. Los roles sobre la gestión de riesgo en el proceso de desarrollo de productos digitales en MS4M deberán estar agrupados en tres niveles. El primero es el estratégico, representado por la Gerencia General quien aprueba las políticas, preside los comités, asigna recursos, asigna objetivos y metas, y supervisa el cumplimiento de los lineamientos base de la gestión de riesgos. El segundo es el táctico, representado por el gerente y el arquitecto de seguridad de la información, quien preside los lineamientos técnicos de las medidas de control de generación de brechas de seguridad, diseñan y supervisan dichas medidas y las mantienen vigentes y actualizadas. En

el nivel operativo se encuentran los gerentes de desarrollo de productos digitales incluyendo a desarrolladores y arquitectos, que son responsables de implementar los controles diseñados que incluye la identificación, evaluación y control de los riesgos inherentes al desarrollo.

Procesos para identificar, evaluar y controlar los riesgos asociados al desarrollo de software. MS4M debe implementar procesos formales de gestión de riesgos de forma cualitativa o cuantitativa, de forma tal que puedan ser gestionados antes de generar impactos a sus clientes y así misma. Estos procesos deben incluir la valoración cualitativa de la probabilidad de ocurrencia de eventos, tal cual se hace en la valoración de los riesgos de MS4M según su sistema de gestión de calidad, conforme se muestra en la siguiente tabla.

Tabla 52

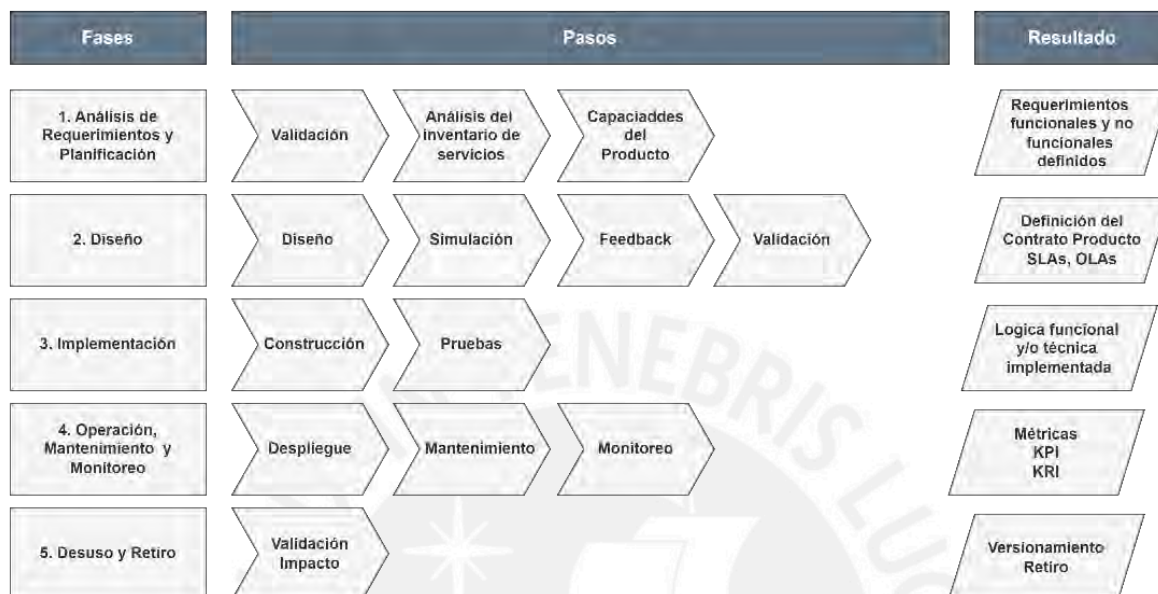
Probabilidad de riesgos

<i>Valoración cualitativa</i>	<i>Escala</i>	<i>Descripción</i>	<i>Frecuencia</i>
Rara vez	1	Puede que no se haya presentado u ocurrido sólo en circunstancias excepcionales.	Nunca o no se ha presentado en los últimos 5 años
Improbable	2	Puede ocurrir en algún momento, es poco común o frecuente	Al menos una vez en los últimos 5 años
Posible	3	Puede ocurrir en algún momento	Al menos una vez en los últimos 2 años
Probable	4	Ocurrirá en la mayoría de las circunstancias	Al menos una vez en el último año
Casi seguro	5	Se espera que ocurra en la mayoría de las circunstancias	Más de una vez al año

Nota. Tomado de SIG-PRO-02 Gestión de Riesgos y Oportunidades. MS4M.

Criterios de evaluación de los impactos. De la misma forma, MS4M debe valorar el posible impacto de las brechas de seguridad y calidad ocasionados durante el proceso de desarrollo de productos digitales alineados a la gestión de riesgos corporativa, debiendo usar los criterios establecidos en SIG-PRO-02 gestión de riesgos y oportunidades de su sistema de gestión de calidad.

- Implementación de medidas de mitigación y control. Estas medidas tendrán tres dimensiones, la primera relacionada a controles a implementarse antes de un posible evento, de carácter preventivo que incluye controles administrativos como políticas, normas contractuales, procedimientos y estándares y controles de ingeniería como doble autenticación, controles de cantidad y tiempos de conexión, canales seguros de conexión, cifrados y encriptación, y otros relevantes a la preservación de la integridad y utilización del código. La segunda está involucrada a controles durante un posible incidente, tales como cifrado de comunicaciones, aislamiento de sistemas comprometidos, gestión de parches emergentes u otros expuestos a la conectividad a internet. La tercera es de mitigación, después de ocurrido el incidente, como investigaciones formales o forenses, procesos de roll-back, redistribución de permisos y privilegios, entre otras. Todas ellas en conjunto deberán de estar integradas y alineadas a la gestión de riesgos corporativa.
- Integración a través de todo el ciclo de desarrollo de software. MS4M debe establecer las pautas del modelo de desarrollo de productos digitales e integrar los controles a dicho ciclo. El modelo operacional se muestra en la siguiente figura 21.

Figura 21*Ciclo de vida del desarrollo de productos digitales*

Nota: Elaboración propia

En ella se resalta la necesidad del levantamiento de los requisitos no funcionales como parte del alcance inicial y la planificación. De la misma forma la inclusión de riesgos a identificarse desde las etapas iniciales.

- Alineamiento con requisitos legales y normativos del mercado. La gestión de riesgos en el desarrollo de productos digitales debe estar alineado principalmente a dos regulaciones, la primera es la Ley 29733 de Protección de datos personales, y la segunda de carácter voluntario, ISO 27000:2022.

Respecto al cumplimiento normativo, MS4M debe asegurar que su proceso de desarrollo de productos digitales tenga los siguientes lineamientos.

- Identificación de normas aplicables desde tres perspectivas como requisitos gubernamentales, del mercado y de estándares corporativos de los clientes.
El marco legal que corresponde para MS4M se concentran en el Decreto legislativo 822 – Ley sobre derechos de autor, Ley 29733 de Protección de datos personales. Ley 30309 de promoción de la inversión privada en el desarrollo e innovación tecnológica, normas voluntarias como ISO 27000:2022 y CMMI Modelo integrado de madurez de capacidades.
- Implementando capacidades de negocio, para asegurar controles de seguridad y cumplir con los requisitos normativos, incorporando los servicios de consultoría legal, realización de auditorías para verificar el cumplimiento, capacitación y concientización sobre el marco regulatorio, gestión ante incidentes y violaciones del marco regulatorio y sobre todo, mejora continua en el proceso.

Estos lineamientos base permitirán a MS4M iniciar el proceso de despliegue del marco de referencia. Sin embargo, existen otros lineamientos a desarrollarse tomando en cuenta un crecimiento inminente.

- Capacitación, entrenamiento y concientización
- Comunicaciones
- Promoción de una cultura de colaboración

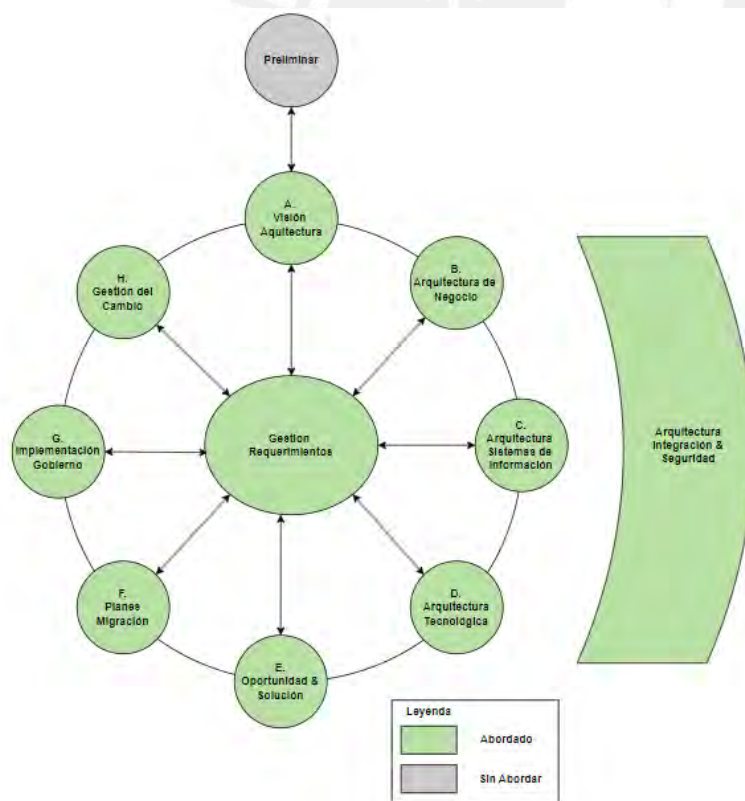
7.2. Modelo de arquitectura empresarial para el desarrollo de software

El objetivo es definir un marco de arquitectura empresarial para gestionar el modelo operativo de productos digitales en MS4M. Para ello, se define el siguiente alcance para ser tratado de manera inmediata:

Método de desarrollo de arquitectura empresarial. Se propone un Método de Desarrollo de Arquitectura Empresarial basado en el ADM (Architecture Development Method) del marco de arquitectura empresarial TOGAF, en el cual se han abordado algunas fases, las cuales se muestran en la figura 22:

Figura 22

Método de desarrollo de arquitectura empresarial



Nota. Adaptado de The Open Group. (2022). TOGAF® 10.^a edition . The Open Group Standard. Recuperado de <https://www.opengroup.org/togaf>

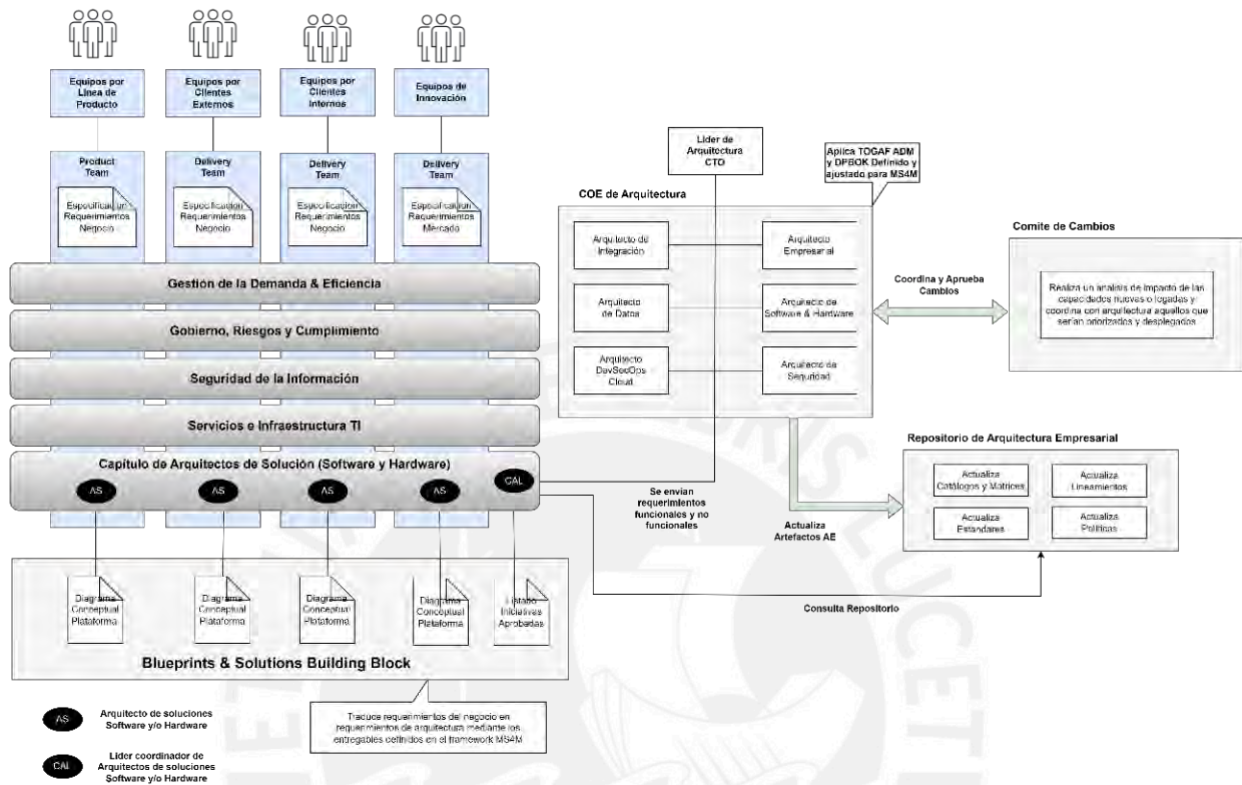
Es importante tener en cuenta que los dominios de arquitectura de desarrollo y seguridad son transversales a los dominios de negocio, sistemas de información y arquitectura tecnológica. El esquema propuesto se centra en la atención de nuevos requerimientos y evolutivos, y tiene por objetivo mejorar la eficiencia operativa y la experiencia de los clientes internos y externos de MS4M.

Para ello se ha tomado como recomendación una estructura organizacional de tipo hélice para la entrega de valor a nivel de arquitectura en los productos digitales. Se puede apreciar en la figura 23 la transversalidad de los procesos de gestión de requerimientos soportada por una arquitectura de building blocks con centros de excelencia especializados en los diseños y entregas.

Las operaciones de MS4M en dimensión representan a una mediana empresa y la presente propuesta ha sido dimensionada para el volumen actual de requerimientos, una mayor demanda o un crecimiento con nuevas capacidades de negocio demandarán una nueva arquitectura o nuevos elementos.

Figura 23

Flujo de atención de requerimientos de desarrollo



Nota. Elaboración propia.

En este modelo, los requerimientos provienen de cuatro fuentes, como evolución de un producto, por clientes externos, internos o de actividades de innovación; el Arquitecto de Soluciones traduce los requisitos de negocio en requisitos de arquitectura y entregan diagramas conceptuales y de infraestructura de la solución (lógicos y físicos); para luego validarlos en el repositorio de arquitectura, los primeros entregables pasan a la categoría de validación. Aquí el Arquitecto de Negocio valida que la solución esté alineada al mapa de capacidades de negocio, mide el impacto y actualiza el catálogo de capacidades de negocio si fuese necesario, entregando una versión actualizada de los conceptuales generados. Luego, los Arquitectos de Datos validan el alineamiento con los lineamientos de arquitectura y

modelado, además de medir el impacto de modificaciones o creaciones de unidades nuevas, entregando una nueva versión actualizada de los conceptos y el catálogo de entidades actualizado. Con estas entradas, el Arquitecto de Aplicaciones valida que las funcionalidades que se van a implementar no estén duplicadas en otros productos digitales, entregando una nueva versión de los conceptos, además del catálogo de portafolio de productos y una matriz de producto/función actualizada. Con estas nuevas entradas, el Arquitecto de Tecnología revisa que la arquitectura del producto digital respete los lineamientos y estándares definidos, además de evaluar cada componente tecnológico para evaluar el impacto y medir la obsolescencia tecnológica, entregando luego una nueva versión de los conceptos y el catálogo de componentes actualizado. Luego de esta etapa, un comité de cambios actualiza las versiones autorizando las variantes para luego actualizar los repositorios. Otro elemento del alcance son los principios de arquitectura que gobiernan los procesos. En la siguiente tabla 52 se resumen.

Tabla 53*Principios de la arquitectura*

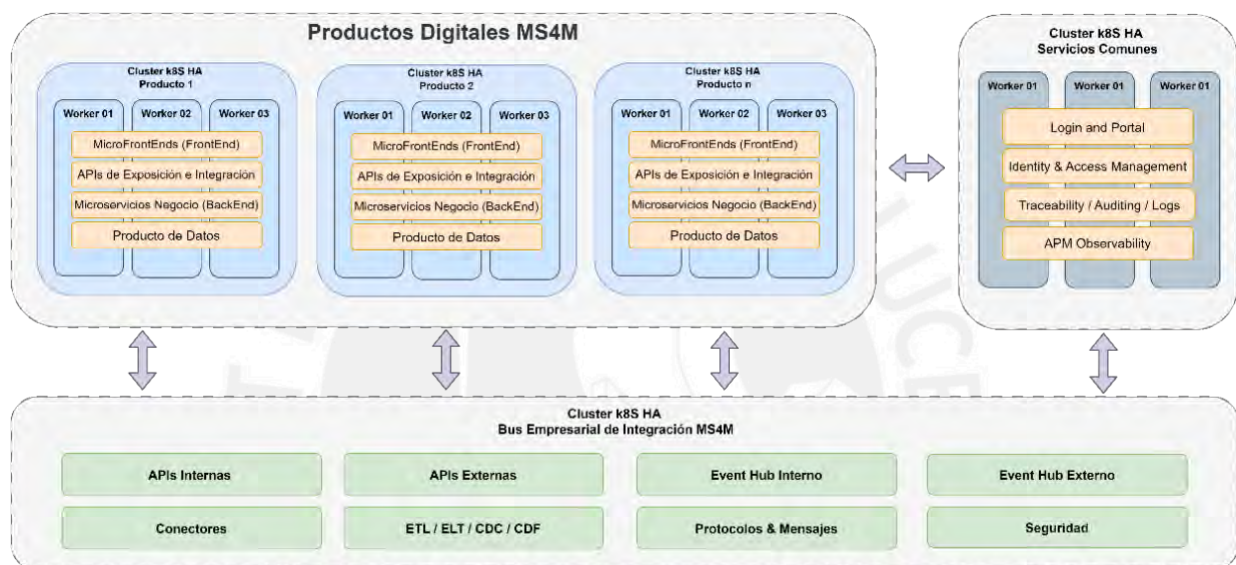
<i>Principio</i>	<i>Descripción</i>
Escalabilidad / Elasticidad	Desarrollar productos bajo estándares y tecnologías orientadas a nubes híbridas (SaaS, Serverless, PaaS) que corran propiamente en cualquier proveedor de nube u On-Premise, y que garanticen soportar altas cargas de procesamiento paralelo y que sean soluciones elásticas end-to-end que permitan aumentar ante alta carga y disminuir ante la ausencia de ésta de manera automática.
Personalización / Configurabilidad	Construir productos que estén diseñadas y construidas para ser altamente configurables y flexibles (versus estáticas)
Conectividad/Reusabilidad	Implementar productos que resuelvan capacidades de negocio o técnicas y que se puedan exponer interna y externamente mediante APIs para que puedan ser llamadas por diferentes productos y/o aplicaciones.
Disponibilidad / Confiabilidad	Construir productos con alto uptime (99.9%) que garanticen que las aplicaciones estén operativas 24/7.
Eficiencia	Implementar productos con diseños en costos eficientes, basados en buenas prácticas, reduciendo los costos operativos y tareas rutinarias como la gestión de los servidores.
Mantenibilidad	Implementar productos con diseños y desarrollos bien estructurados que permitan modificar/adicionar rápidamente componentes sin afectar a los demás. Priorizar soluciones empresariales gestionadas (SaaS, PaaS) que demanden un esfuerzo operacional bajo a la organización.
Sensorización	Fomentar la recolección y uso de información embebidas en los productos, aplicaciones y su hardware para entender mejor a los usuarios / clientes, y sugerirles en tiempo real opciones (por ejemplo, servicios adicionales, mejora de configuración, etc.).
Seguridad	Construir productos diseñados desde un principio con altos estándares de seguridad para su implementación y uso.
Valor en corto plazo	Priorizar opciones tecnológicas que generen valor a corto plazo al negocio a la hora de tomar decisiones en temas de plataformas, componentes y herramientas.
Desacoplamiento	Diseñar productos que permitan separar responsabilidades de transacciones y consultas en plataformas independientes y específicas.

Nota. Elaboración propia

Otro elemento del alcance es la arquitectura del producto digital a desarrollarse. Se propone un conjunto de tres capas con un objetivo específica en cada una, las cuales se muestran en la siguiente figura 24.

Figura 24

Arquitectura de un producto digital en MS4M.



Nota. Elaboración propia. No publicado

La capa de productos digitales de MS4M contendrá las conexiones para cada producto digital construido en MS4M, contendrá las funcionalidades de negocio y experiencia ofrecidas, además de habilitar las capacidades de exposición para poder habilitar las comunicaciones síncronas y asíncronas entre todos los productos que se conecten a esta capa. La capa de servicios comunes contendrá las capacidades transversales para que los productos digitales puedan trabajar. Las capacidades que habilita son el acceso y portal menú de acceso a las funcionalidades de cada producto digital, el componente que gestiona las identidades de los usuarios y dispositivos, el componente que maneja la trazabilidad de las transacciones y

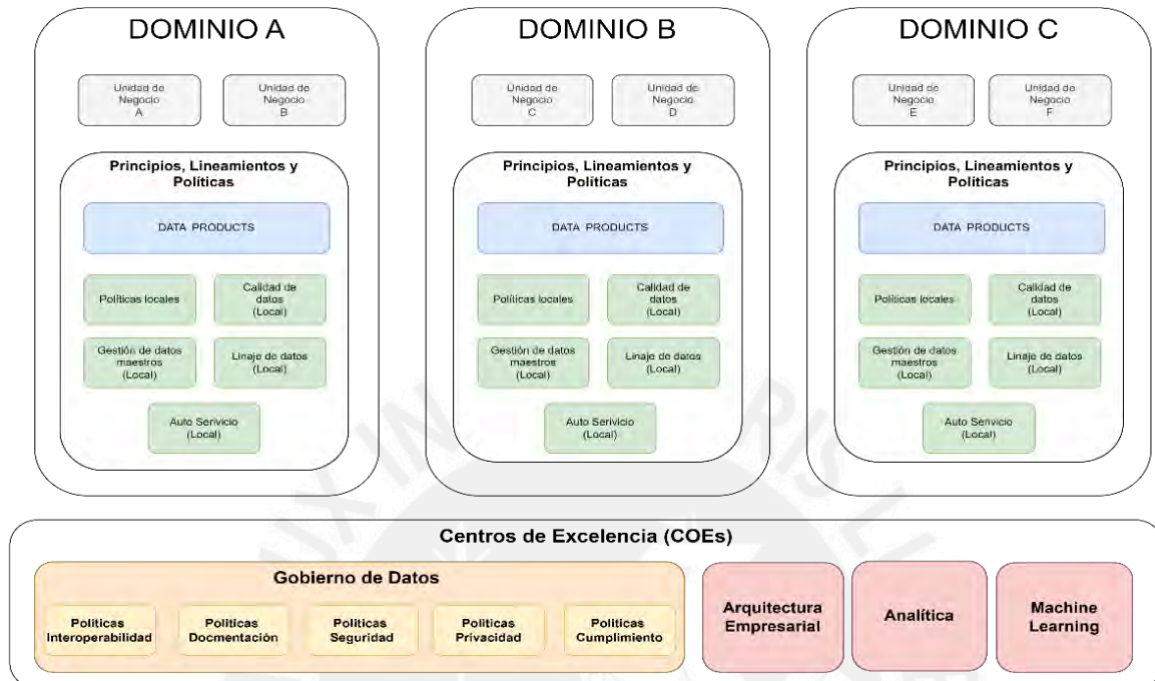
el componente de gestor de rendimiento para monitorear la salud de los servicios e infraestructura. La última capa es la del bus empresarial de integración, que permitirá integrar todos los productos digitales MS4M, integrarse con soluciones de terceros, manejar distintos protocolos y la seguridad a nivel de los servicios, mensajes y protocolos.

Otro elemento del alcance es la arquitectura de datos. Se propone un modelo de gobierno federado basado en dominios y subdominios como se muestra en la figura 25, bajo los siguientes estilos de arquitectura.

- Se aplicarán estilos de gobierno y arquitectura de datos basados Data Mesh con Data Fabric, se utilizarán lo mejor de ambos estilos.
- La unidad mínima de gobierno establecida dentro de un dominio será un producto de datos, esto con la finalidad de tener un buen gobierno, mejor calidad y catalogación de los datos.
- A nivel de arquitectura tecnológica se adoptarán patrones de *Data Lake House*, con la finalidad de adoptar arquitecturas modernas, fácilmente escalables, y ágiles.
- A nivel de flujos y orquestación de datos se construirán flujos de tipo DataOps y MLOpS para mejorar los procesos de catalogación y linaje de los datos.
- Se requiere personal altamente especializado.
- Estructura organizacional a altura de cada unidad de negocio.

Figura 25

Gobierno operativo federado

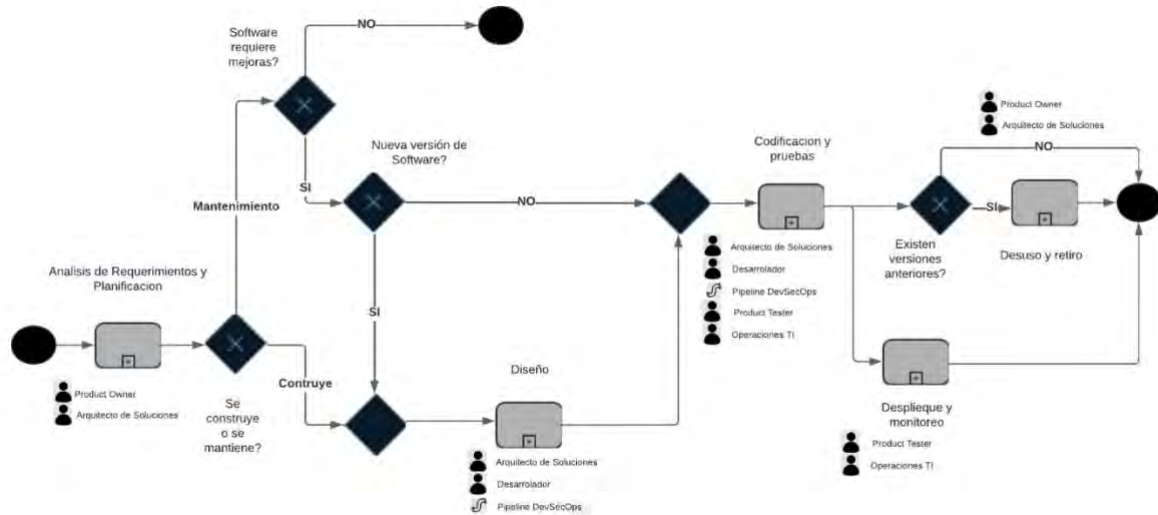


Nota. Elaboración propia. No publicado

A continuación, se muestran los procesos de atención de requerimientos de desarrollo. En la figura 26 se detalla el ciclo de vida de los productos digitales, incluido software, a desarrollarse para una mejor gestión. En ella se muestran de forma simplificada 5 procesos inherentes dentro de este ciclo de vida. La primera relacionada a la evaluación de requerimientos, el diseño, la implementación, la operación y el desuso.

Figura 26

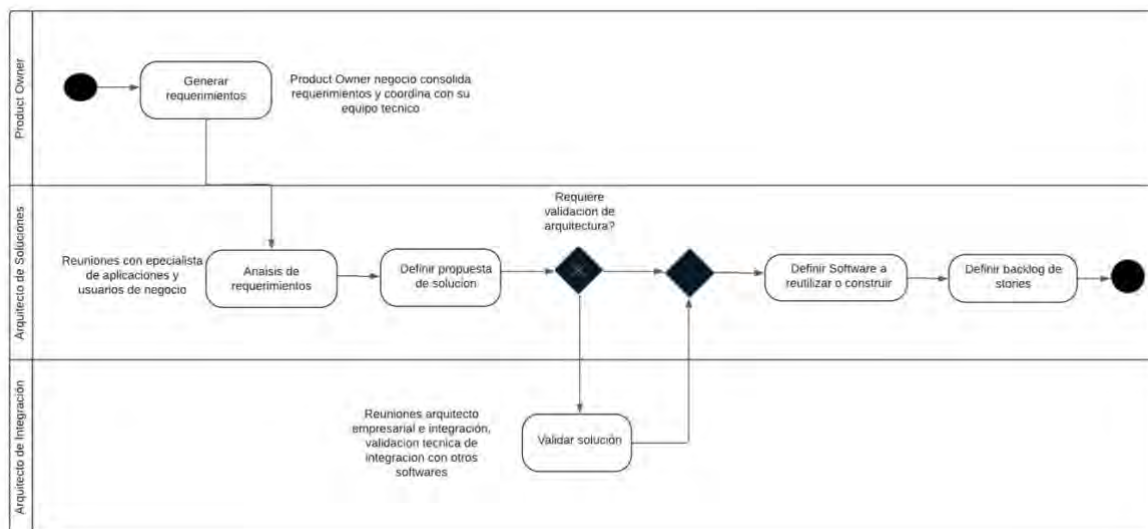
Proceso recomendado del ciclo de vida del desarrollo de un producto de software (Nivel 1)



Nota. Elaboración propia.

Figura 27

Proceso recomendado de análisis de requerimiento y planificación de productos de software (Nivel 2)

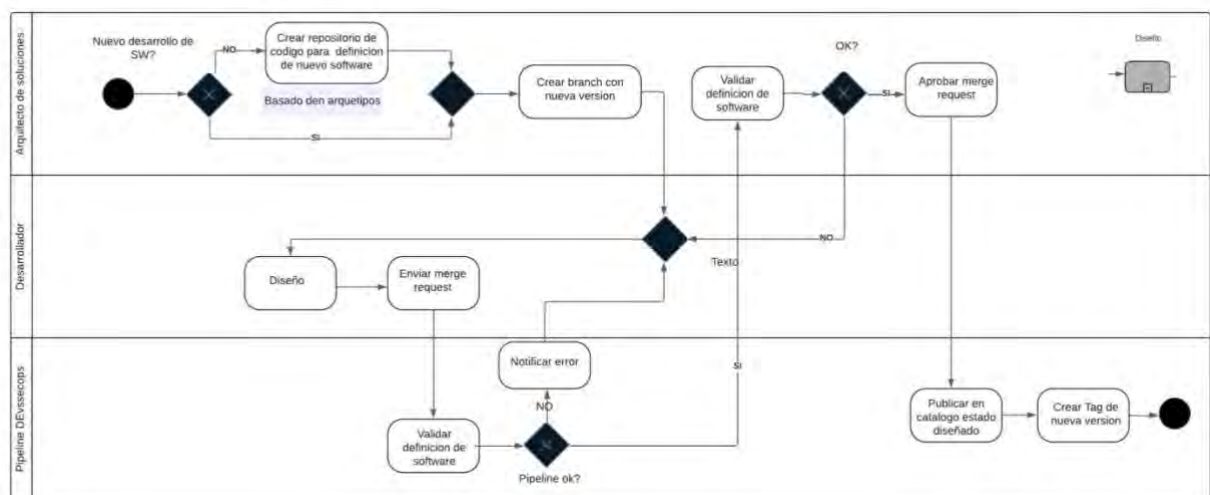


Nota. Elaboración propia.

La figura 27 muestra el proceso donde se inicia la producción de productos digitales, se toman dos decisiones principales, la primera respecto al alineamiento con el negocio, aquí el arquitecto interpreta las necesidades del negocio y las traduce en funcionalidades y lenguaje de desarrollador, en donde el entendimiento de las reglas del negocio y la comunicación con el dueño del producto son fundamentales. La segunda es la compatibilidad con el resto de los sistemas, en donde la salida principal es el concepto validado. Culmina con el almacenamiento en el repositorio para esta etapa.

Figura 28

Proceso de diseño de productos de software (Nivel 2)

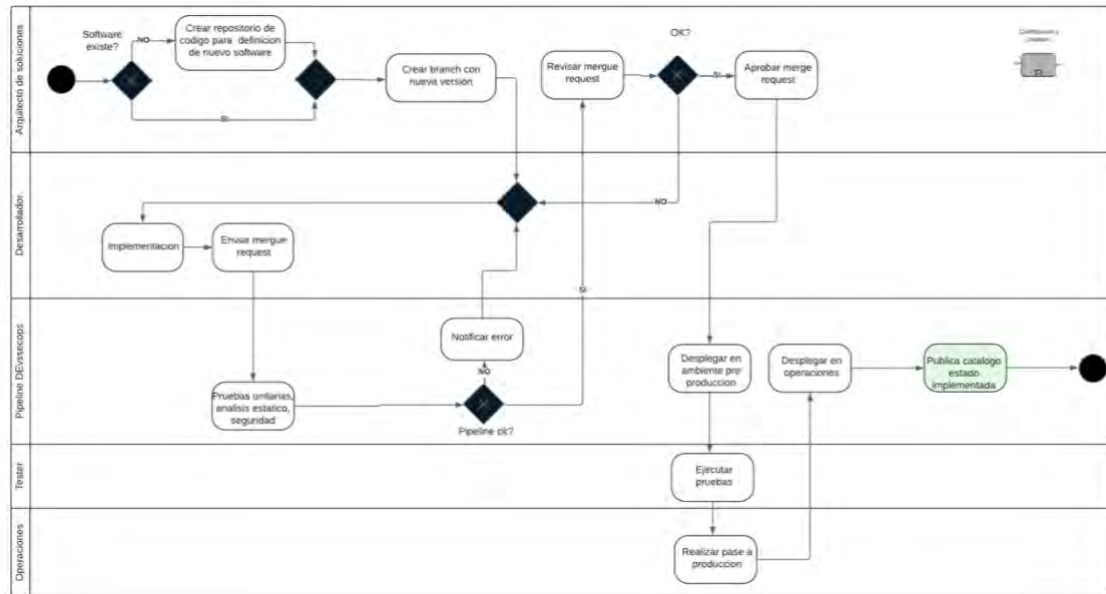


Nota. Elaboración propia.

En esta etapa mostrada en la figura 28, los conceptos son sometidos a pruebas de código por el desarrollador y son validadas por el arquitecto, cuya función sigue siendo el alineamiento del código generado con las reglas del negocio. La salida es la publicación en el repositorio de esta etapa.

Figura 29

Proceso de implementación de productos de software (Nivel 2)

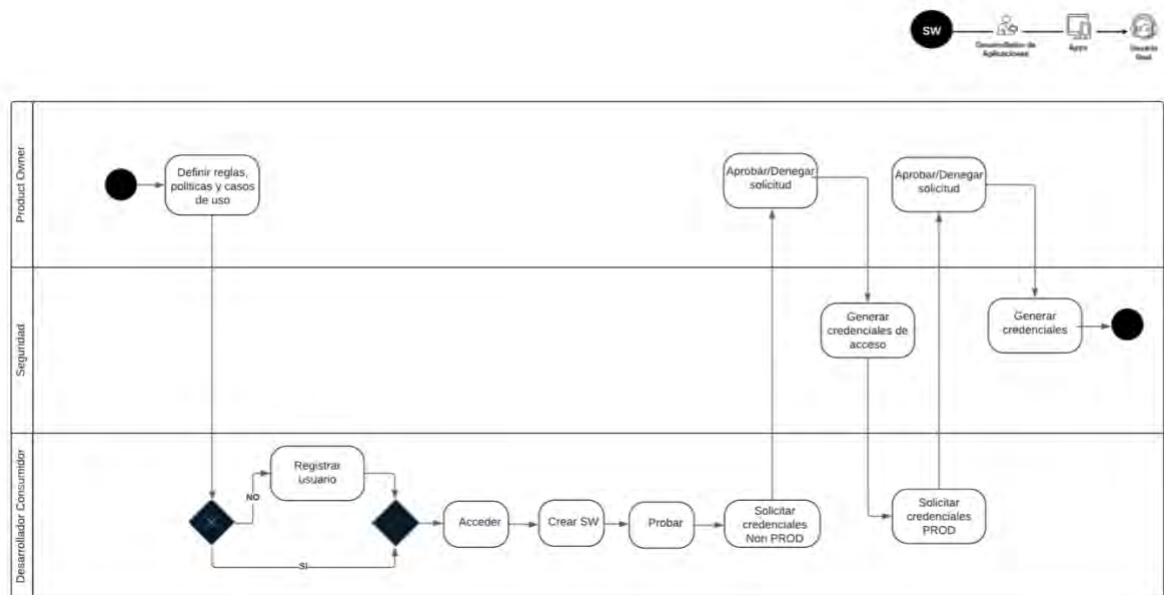


Nota. Elaboración propia.

La figura 29 muestra el proceso de implementación en donde el arquitecto primero decide y valida la existencia para evitar duplicidad y ejecuta funciones de control de versiones; el ingeniero de control de calidad somete a prueba en el ambiente controlado antes del pase a producción luego de ello el especialista en DevSecOps hace el despliegue y almacena en el catálogo de esta etapa.

Figura 30

Proceso de operación de productos de software (Nivel 2)

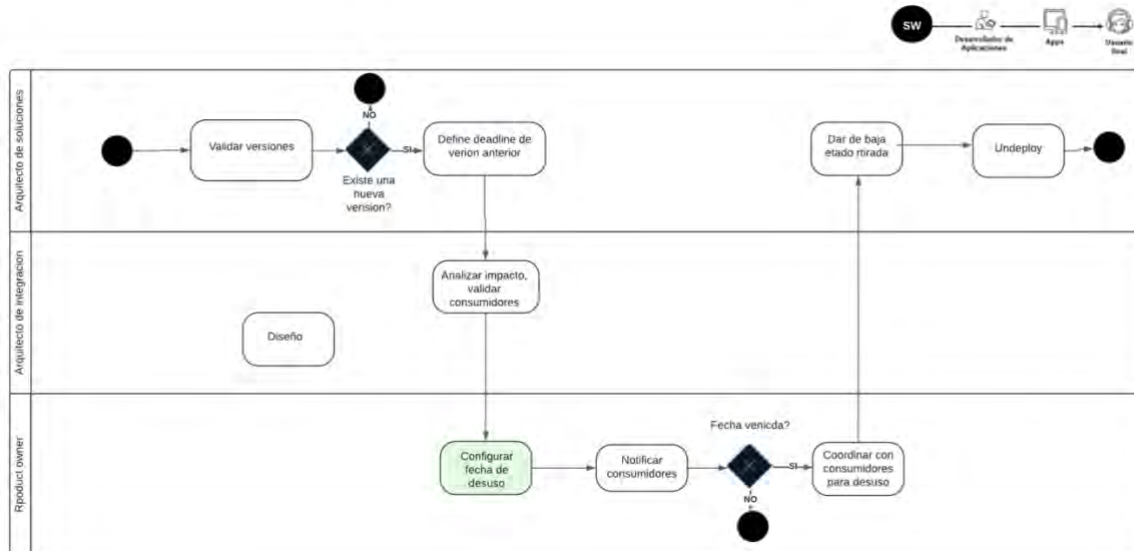


Nota. Elaboración propia.

Esta etapa mostrada en la figura 30 muestra la intervención del especialista de seguridad de la información para el control de accesos al producto digital creado. La función principal del dueño del producto radica en la documentación de los manuales de usuario acorde a las reglas de negocio sobre las cuales se desarrolló el producto.

Figura 31

Proceso de desuso de productos de software (Nivel 2)



Nota. Elaboración propia.

Tan importante como la puesta en operación es el control de versiones cuando se actualicen, aquí el arquitecto de soluciones realiza las definiciones y el arquitecto de integraciones mide el impacto antes de la actualización en el resto del ecosistema. Este proceso permite un catálogo ordenado fuente para cualquier otra iniciativa.

Otros elementos del alcance deben de ser desarrollados/adaptados en forma más detallada a niveles de usuario con personal involucrado de MS4M, entre ellos están:

- Visión de la arquitectura empresarial, que incluye además de los principios, lineamientos y estándares que deben ser cumplidos por el producto digital a desarrollarse.
- Glosario unificado y estándar de términos.

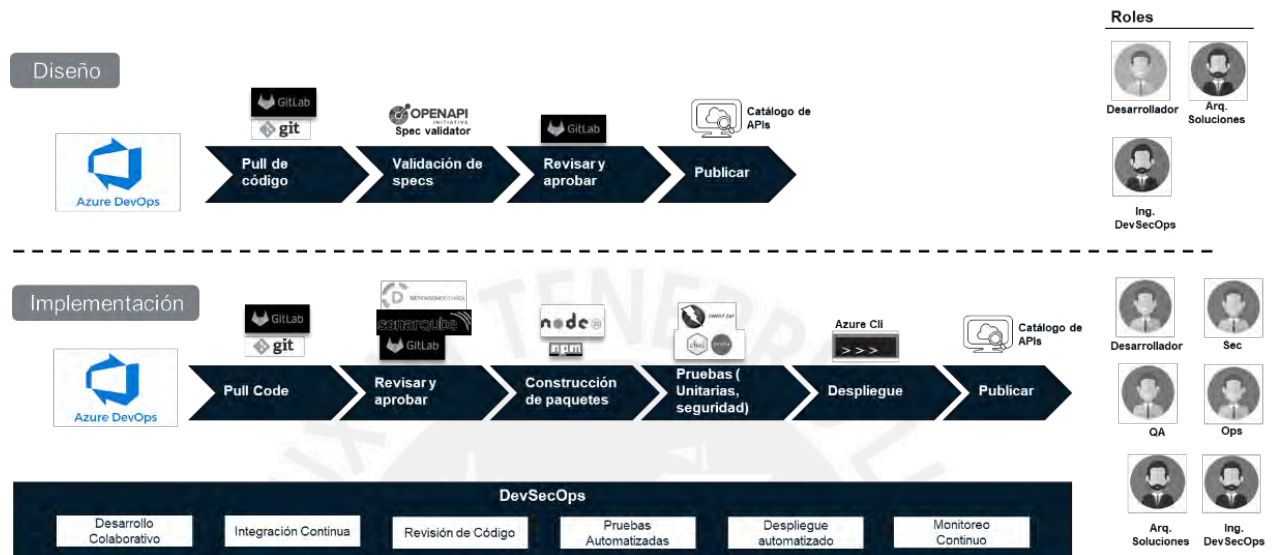
- La arquitectura de negocio con los artefactos talas como las capacidades de negocio descritas por los propios miembros.
- Dentro de la arquitectura de datos, los lineamientos de los estilos Data Mesh y Data Fabric, acorde a su necesidad de mercado.
- La arquitectura de integración a nivel de línea base
- La arquitectura tecnológica donde se describan los componentes tecnológicos que soportan los procesos de desarrollo de productos digitales.
- Dentro del gobierno de la arquitectura, la definición de roles y responsabilidades a niveles estratégicos y operativos.
- El repositorio de la arquitectura, donde se indique la estructura y clases de información que almacenará.
- Modelo de financiamiento, donde se incluyan las metodologías para incluir los costos operativos y de inversión en el desarrollo del producto digital.

7.3. DevSecOps como marco de referencia para el desarrollo de software

MS4M deberá contar con un manual de buenas prácticas para el desarrollo de software en MS4M basados en la metodología de DevSecOps, siguiendo el siguiente flujo:

Figura 32

Flujo de integración y entrega continua



Nota. Elaboración propia. No publicado.

La Figura 32 explica el proceso de diseño e implementación en base a DevSecOps. Se sugiere desde el proceso de diseño, el alineamiento a través del Arquitecto de Soluciones, de los requerimientos a las necesidades y reglas de negocio desde el repositorio de código hasta una posterior publicación, en donde en la etapa de implementación a través de control de calidad se realicen las pruebas unitarias en los ambientes controlados disponibles para un posterior despliegue. Las revisiones difieren en ambas etapas a alineamientos diferentes, la primera es respecto a las reglas de negocio, la segunda es respecto a la calidad.

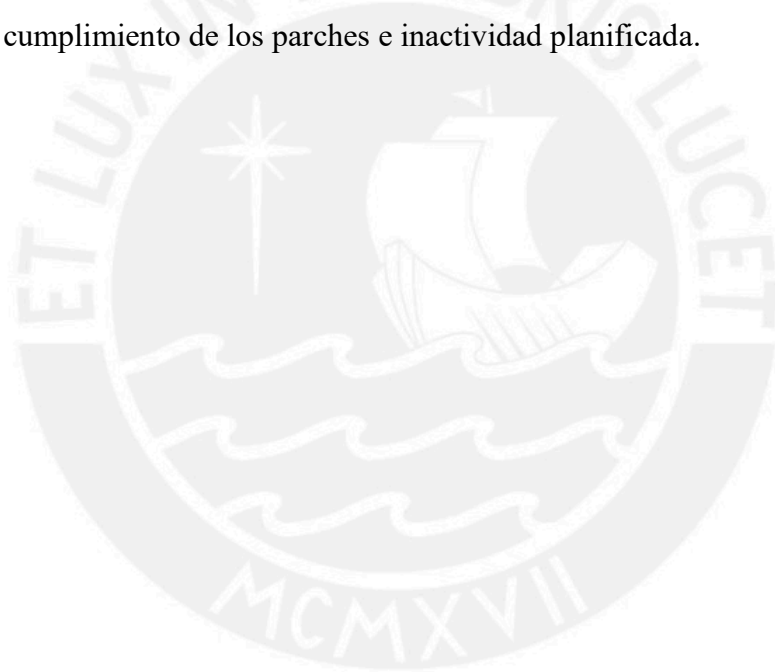
Complementario a ello, los siguientes principios y prácticas base de DevSecOps son base de los procesos de diseño e implementación que deben de desarrollarse e implementarse:

- Desarrollo colaborativo. Se deben de aplicar las políticas de control de versiones, estrategia de ramas para uso de repositorios, flujos de aprobación

y revisión del código estático, pero sobre todo divisiones de tareas y responsabilidades a nivel pequeño de mínima unidad.

- Integración continua. Aplicar las políticas de detección de cambios en los repositorios del código fuente, creación de los flujos de despliegue, compilación del código fuente y vinculación de librerías usadas, ejecutar pruebas automatizadas, registrar paquetes nuevos y actualizados en el repositorio de administración de paquetes, para implementarlos en entornos posteriores.
- Revisión de código. Aplicar las políticas de código seguro, como mensajes de error más generalizados para evitar envío de información relevante al usuario y generar brechas.
- Pruebas automatizadas. Se deben aplicar las políticas de automatización de las pruebas de código estático (SAST) usando herramientas como SonarQube o Checkmarx para analizar el código fuente en busca de vulnerabilidades; pruebas de código dinámico (DAST) usando herramientas como OWASP ZAP o Burp Suite para realizar pruebas de penetración; revisión de imágenes a través de Calir o Anchore Engine para identificar vulnerabilidades en los paquetes y bibliotecas de la imagen, y revisión de librerías de terceros (SCA) con herramientas como Snyk para gestionar las dependencias de terceros.
- Despliegue automatizado. Aplicar las políticas de gestión de la configuración y promoción entre los ambientes definidos en el ciclo de desarrollo de productos, tales como la habilitación de los entornos de pruebas alterno a los de desarrollo y producción, o uso de herramientas como Ansible para automatizar la configuración.

- Monitoreo continuo. Se deben construir reportes de monitoreo con indicadores de rendimiento de la salud del producto y de todos los componentes tecnológicos que lo soportan. Indicadores asociados a la disponibilidad del sistema tales como el tiempo medio entre fallas (MTBF) y de reparación (MTTR), asociados al rendimiento del sistema como tiempos de respuesta medio, tasas de transferencia de datos o de usos de recursos, asociados a la capacidad como los límites y umbrales de usuarios concurrentes, y asociadas a mantenimiento y actualizaciones como cumplimiento de los parches e inactividad planificada.



Capítulo VIII: Plan de Implementación y Factores clave de éxito

En este capítulo se detallará las acciones necesarias para llevar a cabo la implementación de la solución propuesta, considerando los recursos involucrados, la duración de las actividades y los costos involucrados en el presente proceso, a continuación, se listan las actividades identificadas:

8.1. Actividades

Se tiene el siguiente listado de actividades de la implementación de la solución propuesta.

Tabla 54

Actividades de la solución propuesta (parte 1)

<i>Actividades</i>
Implementación de marcos de referencia para desarrollo de software
Fase de Inicio
Definir los objetivos y alcance del proyecto
Identificar los stakeholders y formar el equipo de proyecto
Realizar un análisis de la situación actual de la organización
Fase de Planificación
Planificar la capacitación del equipo en metodologías ágiles, arquitectura empresarial y gobernanza organizacional
Fase de Ejecución
Análisis inicial
Investigación de mercado y evaluación de necesidades
Identificación de puntos fuertes y diferenciadores
Evaluación de prácticas, herramientas e infraestructura
Planificación y diseño:
- Definición de políticas y procedimientos iniciales
- Selección de herramientas y tecnologías
- Diseño de arquitectura empresarial (AS IS)
Implementación de cambios
Establecimiento de lineamientos y políticas
Implementación de prácticas ágiles y DevSecOps
Diseño de arquitectura empresarial futura (TO BE)
- Evaluación y selección de tecnologías y herramientas
- Desarrollo de procedimientos de implementación y migración
- Planificación de pruebas y validación del nuevo diseño
Integración de metodologías ágiles y DevSecOps
- Revisión y adaptación de flujos de trabajo y roles
- Capacitación adicional para el equipo
Establecimiento de métricas y KPIs

Nota. Elaboración propia.

Tabla 55*Actividades de la solución propuesta (parte 2)*

<i>Actividades</i>
Desarrollo, seguimiento y optimización <ul style="list-style-type: none"> - Gestión y modelado del ciclo de vida del software - Seguimiento y gestión del proyecto - Atención de incidencias de nivel 0, 1 y 2 - Finalización y cierre formal de incidencias - Documentación de lecciones aprendidas - Análisis post-mortem para identificar mejoras
Elaboración de documentación: <ul style="list-style-type: none"> - Desarrollo de políticas, lineamientos y plantillas - Revisión y aprobación de la documentación
Evaluación y ajuste: <ul style="list-style-type: none"> - Revisión continua del progreso del proyecto - Ajuste de estrategias y recursos según sea necesario
Fase de Monitoreo y Control (durante toda la ejecución) <ul style="list-style-type: none"> Recoger feedback y realizar ajustes necesarios durante la implementación (continuo). Realizar auditorías periódicas para asegurar el cumplimiento de las nuevas políticas y procedimientos (continuo).
Fase de Cierre <ul style="list-style-type: none"> Realizar una revisión post-implementación Identificar áreas de mejora y actualizar los procesos y políticas

Nota. Elaboración propia

8.2. Diagrama de Gantt de Implementación

Se muestra el diagrama de Gantt de Implementación de marcos de referencia.

Tabla 56*Diagrama de Gantt (parte 1)*

<i>Actividades</i>	<i>Duración</i>	<i>Comienzo</i>	<i>Fin</i>
Implementación de marcos de referencia para desarrollo de software	184 días	lun 3/06/24	jue 13/02/25
Fase de Inicio	10 días	lun 3/06/24	vie 14/06/24
Definir los objetivos y alcance del proyecto	3 días	lun 3/06/24	mié 5/06/24
Identificar los stakeholders y formar el equipo de proyecto	3 días	jue 6/06/24	lun 10/06/24
Realizar un análisis de la situación actual de la organización	4 días	mar 11/06/24	vie 14/06/24
Fase de Planificación	8 días	lun 17/06/24	mié 26/06/24
Planificar la capacitación del equipo en metodologías ágiles, arquitectura empresarial y gobernanza organizacional	8 días	lun 17/06/24	mié 26/06/24
Fase de Ejecución	176 días	lun 3/06/24	lun 3/02/25
Análisis inicial	30 días	jue 27/06/24	mié 7/08/24
Investigación de mercado y evaluación de necesidades	10 días	jue 27/06/24	mié 10/07/24
Identificación de puntos fuertes y diferenciadores	5 días	jue 11/07/24	mié 17/07/24
Evaluación de prácticas, herramientas e infraestructura	15 días	jue 18/07/24	mié 7/08/24
Planificación y diseño:	25 días	jue 8/08/24	mié 11/09/24
Definición de políticas y procedimientos iniciales	5 días	jue 8/08/24	mié 14/08/24
Selección de herramientas y tecnologías	10 días	jue 15/08/24	mié 28/08/24
Diseño de arquitectura empresarial (AS IS)	10 días	jue 29/08/24	mié 11/09/24
Implementación de cambios	56 días	lun 3/06/24	lun 19/08/24
Establecimiento de lineamientos y políticas	10 días	lun 3/06/24	vie 14/06/24
Implementación de prácticas ágiles y DevSecOps	15 días	lun 17/06/24	vie 5/07/24
Diseño de arquitectura empresarial futura (TO BE)	20 días	lun 8/07/24	vie 2/08/24
- Evaluación y selección de tecnologías y herramientas	5 días	lun 8/07/24	vie 12/07/24
- Desarrollo de procedimientos de implementación y migración	10 días	lun 15/07/24	vie 26/07/24
- Planificación de pruebas y validación del nuevo diseño	5 días	lun 29/07/24	vie 2/08/24
Integración de metodologías ágiles y DevSecOps	10 días	lun 5/08/24	vie 16/08/24
- Revisión y adaptación de flujos de trabajo y roles	5 días	lun 5/08/24	vie 9/08/24
- Capacitación adicional para el equipo	5 días	lun 12/08/24	vie 16/08/24
Establecimiento de métricas y KPIs	1 día	lun 19/08/24	lun 19/08/24

Nota. Elaboración propia.

Tabla 57*Diagrama de Gantt (parte 2)*

<i>Actividades</i>	<i>Duración</i>	<i>Comienzo</i>	<i>Fin</i>
Desarrollo, seguimiento y optimización	111 días	lun 3/06/24	lun 4/11/24
Gestión y modelado del ciclo de vida del software	30 días	lun 3/06/24	vie 12/07/24
Seguimiento y gestión del proyecto	20 días	mar 20/08/24	lun 16/09/24
Atención de incidencias de nivel 0, 1 y 2	15 días	mar 17/09/24	lun 7/10/24
Finalización y cierre formal de incidencias	5 días	mar 8/10/24	lun 14/10/24
Documentación de lecciones aprendidas	5 días	mar 15/10/24	lun 21/10/24
Análisis post-mortem para identificar mejoras	10 días	mar 22/10/24	lun 4/11/24
Elaboración de documentación:	40 días	mar 5/11/24	lun 30/12/24
Desarrollo de políticas, lineamientos y plantillas	30 días	mar 5/11/24	lun 16/12/24
Revisión y aprobación de la documentación	10 días	mar 17/12/24	lun 30/12/24
Evaluación y ajuste:	25 días	mar 31/12/24	lun 3/02/25
Revisión continua del progreso del proyecto	15 días	mar 31/12/24	lun 20/01/25
Ajuste de estrategias y recursos según sea necesario	10 días	mar 21/01/25	lun 3/02/25
Fase de Monitoreo y Control (durante toda la ejecución)	184 días	lun 3/06/24	jue 13/02/25
Recoger feedback y realizar ajustes necesarios durante la implementación (continuo).	184 días	lun 3/06/24	jue 13/02/25
Realizar auditorías periódicas para asegurar el cumplimiento de las nuevas políticas y procedimientos (continuo).	184 días	lun 3/06/24	jue 13/02/25
Fase de Cierre	8 días	mar 4/02/25	jue 13/02/25
Realizar una revisión post-implementación	4 días	mar 4/02/25	vie 7/02/25
Identificar áreas de mejora y actualizar los procesos y políticas	4 días	lun 10/02/25	jue 13/02/25

Nota. Elaboración propia.

A continuación, se describirán las consideraciones y modo ejecución del proyecto de implementación de los marcos de referencia en MS4M:

En la fase inicial, de acuerdo a los objetivos definidos y la revisión del estado situacional del presente documento, el equipo consultor en coordinación con MS4M, determinaron como área de análisis al área de Investigación y Desarrollo (I+D) del departamento de tecnologías de la información, de esta manera se analizó directamente el proceso de atención de requerimientos e incidencias (ver Tabla 6 y 7). Asimismo, se consideró como parte del análisis, establecer los proyectos que pueden considerar la implementación inmediata de las buenas prácticas de los marcos de referencias adoptados,

para lo cual MS4M y el equipo consultor consideraron viable la implementación en los proyectos asociados al producto “Control 4Miners” – C4M, listados en la tabla N° 58. Estos proyectos seleccionados se encuentran planificados para su inicio desde su concepción durante la etapa de implementación de los marcos de referencia, de esta manera permitirán conocer el impacto desde el inicio del proceso de desarrollo y conceptualización, soportados en el nuevo proceso de desarrollo y buenas prácticas.

Tabla 58

Listado de proyectos de C4M

<i>N°</i>	<i>Descripción</i>	<i>Meses</i>
1	Payload Bucket System.	4.5
2	Strategic Optimizer.	6.2
3	Personnel geolocalization in surface mining.	6
4	Compaction Module.	4
5	Blending module.	5
6	Tramming.	4.8

Nota. Elaboración propia.

8.2.1. Sobre los Lineamientos, políticas base de gobierno sobre el desarrollo de software

Los proyectos seleccionados cuentan con una iniciativa alineada a la estrategia corporativa, en concordancia con los objetivos comerciales de MS4M. De esta manera, se busca asegurar la inclusión de procedimientos de calidad de código y producto, así como la incorporación de buenas prácticas de seguridad. Esto permitirá que el proyecto adopte el nuevo proceso de desarrollo, el cual deberá aplicar los lineamientos definidos en la construcción de nuevos productos y/o desarrollos.

8.2.2. Sobre el Modelo de arquitectura empresarial para el desarrollo de software

Es importante indicar que, al considerar la gestión de cambios y requerimientos bajo el método de desarrollo de arquitectura de TOGAF, MS4M permitirá integrar el negocio, las

aplicaciones, la arquitectura de negocio y los aspectos de seguridad. Esto garantizará que las funcionalidades solicitadas sean implementadas de manera escalable y evitará la duplicidad de funcionalidades entre los proyectos seleccionados.

8.2.3. Sobre DevSecOps como marco de referencia para el desarrollo de software

En este sentido, considerando el proceso de desarrollo ágil y la seguridad aplicada que brinda DevSecOps, se fomentará el desarrollo colaborativo y la integración continua en los proyectos seleccionados. A través de entregas continuas y despliegues (que serán implementados conforme a la actualización de la infraestructura tecnológica de MS4M), se permitirá que el proceso de desarrollo sea flexible y ágil.

En concordancia a la política de crecimiento profesional y social de MS4M, de acuerdo con la evaluación de habilidades del personal realizada en el Tabla 30, la ejecución de capacitaciones al personal seleccionado permitirá implementar el nuevo modelo de desarrollo en MS4M. Para esto, el equipo consultor propone identificar las necesidades de capacitación a través de una evaluación de competencias necesarias (ver Figura 32), y con ello validar la capacidad del personal para asumir nuevos roles en el proceso de desarrollo de productos. La ejecución del plan de capacitación buscará identificar y reducir las brechas de habilidades en el personal, la cual deberá ser evaluada de manera progresiva a lo largo de aplicación de los marcos de referencia y DevSecOps en el proceso de desarrollo.

El equipo consultor brindará orientación al equipo de MS4M en el proceso de elaboración de los nuevos procedimientos y políticas para el desarrollo de productos digitales, la aplicación de esta documentación, así como la implementación de las practicas ágiles y DevSecOps deberán realizarse de manera iterativa, con el objetivo de estabilizar los nuevos flujos de desarrollo, integración y pruebas en el ciclo de vida de Software en MS4M.

El equipo consultor propone los siguientes indicadores con el objetivo de mantener la evolución progresiva y favorable del plan de implementación, en la tabla 59 se listan los indicadores iniciales definidos por el equipo consultor:

Tabla 59
Indicadores

<i>Indicador</i>	<i>Descripción</i>	<i>Formula</i>
Cumplimiento del Cronograma	Porcentaje de tareas completadas en relación con el cronograma planificado	$Cumplimiento (\%) = \left(\frac{N. Tareas completadas en tiempo}{N. Total de Tareas} \right) * 100$
Cumplimiento de presupuesto	Porcentaje del presupuesto utilizado en relación con el presupuesto total planificado	$Cumplimiento p. (\%) = \left(\frac{Gasto Actual}{Presup.Total} \right) * 100$
Productividad del Equipo	Número promedio de tareas completadas por cada miembro del equipo en un período de tiempo	$Productividad E. = \frac{Num. Tareas completas}{Total Tareas}$
Tasa de Cumplimiento de Políticas (%):	Porcentaje de cumplimiento de las nuevas políticas y procedimientos implementados	$T. Cumplimiento de Políticas(\%) = \frac{Num. Auditorias Superadas}{Total de Auditorias} * 100$

Nota. Elaboración propia.

Es importante que MS4M, asegure el compromiso y cumplimiento de las políticas elaboradas, en este sentido, la ejecución de auditorías de seguimiento u observación del comportamiento del procedimiento permitirá identificar mejoras y correcciones al proceso de implementación, asimismo, la revisión de las estrategias respecto al cumplimiento de la implementación, así como los programas de adquisición de equipos e inversión en la tecnología base deberá ser ajustada conforme se avance en el cronograma.

8.3. Presupuesto

El presupuesto del proyecto evoluciona a partir de las estimaciones que es aplicado a los costos iniciales del proyecto. Luego se desarrolla una línea base de costos. Tener en cuenta que se tiene que elaborar un fondo de contingencias en respuesta de los riesgos o eventos que ocurran.

En la siguiente tabla se ha elaborado el presupuesto para MS4M:

Tabla 60

Presupuesto

<i>Estructura de Costos:</i>	<i>Cantidad</i>	<i>CostoxHora</i>	<i>Horas</i>	<i>Costo</i>	<i>Subtotal</i>
<i>a. Costos de Personal:</i>					
Especialista Devsecops	1	45	800	36,000	
Analista de Seguridad	1	34	800	27,200	
QA	1	34	600	20,400	
Arquitecto de negocios	1	34	600	20,400	
Arquitecto de Soluciones	1	34	500	17,000	
Especialista en Gobierno	1	34	500	17,000	
Desarrollador Senior	1	34	800	27,200	
Ingeniero de Procesos	1	34	800	27,200	
					192,400
<i>b. Costos de Hardware y Software:</i>					
Estaciones de Trabajo	8			8,000	
Soporte				1,000	
Licencias y herramientas				7,000	
					16,000
<i>c. Costos de Capacitación:</i>					
					5,000
<i>2. Estimación de Recursos:</i>					
					3,000
<i>Total del presupuesto. \$/.</i>					216,400

Nota. Elaboración propia.

8.4. Factores clave para el éxito

De acuerdo a la solución propuesta se tiene los siguientes factores claves para el éxito para garantizar el éxito en la implementación de las propuestas de solución:

8.4.1. Adopción efectiva de metodologías ágiles.

Compromiso y participación activa de todos los miembros del equipo en la adopción de prácticas ágiles. Implementación de mecanismos de retroalimentación continua para mejorar y ajustar los procesos ágiles según sea necesario. Capacitación y desarrollo de habilidades en metodologías ágiles para garantizar una comprensión clara y una aplicación efectiva en todo el equipo.

8.4.2. Implementación de políticas de gobernanza sobre el desarrollo de software.

Establecimiento claro de estrategias y objetivos alineados con los objetivos empresariales. Identificación proactiva y gestión de riesgos relacionados con el desarrollo de software. Cumplimiento riguroso de normativas y regulaciones relevantes en el proceso de desarrollo de software. Fomentar una cultura organizacional que promueva la transparencia, la responsabilidad y la ética en todas las etapas del desarrollo de software.

8.4.3. Desarrollo de una arquitectura empresarial sólida.

Definición clara de la estructura organizacional, roles y responsabilidades para garantizar la eficiencia y la colaboración entre los equipos. Establecimiento de procesos clave y prácticas de desarrollo de software estandarizadas para mejorar la coherencia y la calidad del producto final. Provisionamiento de la infraestructura necesaria, incluida la tecnología y los recursos humanos, para respaldar las operaciones de desarrollo de software de manera eficaz y eficiente.

8.4.4. Implementación exitosa de DevSecOps.

Desarrollo y adopción de un manual de buenas prácticas basado en DevSecOps específicamente adaptado a las necesidades y contextos de MS4M. Integración de la seguridad desde el inicio del ciclo de vida del desarrollo de software para garantizar la protección de los activos de la empresa y la satisfacción del cliente. Automatización de procesos y herramientas para aumentar la eficiencia y la consistencia en el desarrollo, la seguridad y las operaciones.



Capítulo IX: Resultados esperados

En el presente capítulo se evaluará la factibilidad económica de la solución propuesta acorde a la realidad operativa y financiera de MS4M descrita en los capítulos anteriores.

9.1. Resultados esperados de la implementación

Para obtener un indicador de la factibilidad económica de implementar lo sugerido, se va a hacer uso del cálculo del valor actual neto junto con la tasa interna de retorno en un periodo de 4 años, debido a la vigencia de su plan estratégico al 2027. Para ello se toma en cuenta los resultados financieros del año 2022 en donde el gasto corresponde a 934,712 dólares anuales en el personal de investigación y desarrollo conforme se muestra en la tabla 18.

El caso se fundamenta en la disminución del tiempo invertido desarrolladores senior que apoyan en el área de soporte cuando la incidencia escala de nivel 1 a nivel 3, ocupando tiempo necesario para el desarrollo de nuevos productos y evoluciones de los actuales, tal como se muestra en la Figura 6. Situación que se ve en la figura 4 donde se muestra la evolución del gasto en el personal y la merma en la utilidad neta de la empresa mostrada en la figura 5, pese a que las ventas anuales se han incrementado tal y como se muestra en la figura 3.

De acuerdo a lo mostrado en la figura 6, el nivel de incidencia del escalamiento al nivel 3 tiene un valor de 20% durante el tiempo de estudio de acuerdo a la tabla 6 y tabla 7, del total tiempo de trabajo de los desarrolladores. La solución propuesta basada en estudios (Chawla, et al, 2023) sugiere y estima una reducción del 50% del tiempo de ingenieros de software aplicando metodologías ágiles, que conservadoramente los autores estimamos que puede representar un 30% y llegar a 14% en las incidencias de escalamiento.

Tabla 61*Estimación de ahorros*

<i>Situación</i>	<i>Actual</i>	<i>Estimado</i>	<i>Ahorro anual</i>
Gasto I&D PEN	3,579,947		
% escalamiento de nivel	20.0%	14%	6%
Costo en soporte N3 PEN	715,989		214,797

Nota. Elaboración propia.

Acorde a la tabla 18, se tiene un gasto anual de 3,579,947 soles equivalentes al 3.85 dólares por sol del año 2022, con un nivel de escalamiento de 20% que representa un gasto en soporte de nivel 3 de 715,989 soles anuales en tiempo de desarrolladores senior dando soporte en lugar de desarrollar nuevos productos o evolucionar los nuevos. El objetivo con las soluciones propuestas es llevarlo a un 14% este escalamiento de forma permanente y como límite, en donde el 6% de diferencia representa 214,797 soles anuales obtenidos de la siguiente fórmula simple:

$$\text{Ahorro en soporte proyectado} = \text{Costo en soporte actual} * \frac{\text{Escalamiento actual} - \text{Escalamiento estimado}}{\text{Escalamiento actual}}$$

Un punto importante a considerar es el mantenimiento y auditoria de la implementación, se ha considerado un 10% anual de la inversión acorde al mercado por los 5 años. Sin embargo, el área comercial de MS4M estima que un mayor tiempo disponible de los desarrolladores para generar nuevos productos puede aportar un 0.2% en las ventas proyectadas en los dos primeros años, un 0.1% en los dos años siguientes y un 0.05% en el último año, debido a la disminución del *time to market*, o el tiempo de lanzamiento de nuevos productos. Este impacto se muestra en la siguiente tabla 50.

Tabla 62*Otros Ingresos*

<i>Tangibles</i>	<i>2024</i>	<i>2025</i>	<i>2026</i>	<i>2027</i>
Ventas en soles	30,498,856	38,300,000	44,045,000	47,875,000
Incremento en las ventas	0.2%	0.1%	0.1%	0.05%
Aporte anual en soles	60,997	38,300	44,045	23,937

Nota. Tomado de *MS4M Strategic Planning 2027*

Acorde al área comercial de MS4M, los ahorros provenientes de las ventas tendrán un efecto al año siguiente de la implementación, por lo que estos ingresos serán considerados sólo en los tres últimos años. De la misma forma, tal como se indica en la tabla 16, MS4M ha desarrollado su plan estratégico al 2027 en donde uno de los objetivos a largo plazo es llegar a un nivel de ventas de 12.5 millones de dólares anuales, y tal como se muestra en los anexos B y C, esto se obtendrá gradualmente año tras año, por lo que los valores de ventas presentados en la tabla 62 desde el año 2025 hacia adelante corresponden a dicho plan. De la misma manera, las ventas relacionadas al 2024 corresponde al presupuesto aprobado para este año.

Estimados los ingresos, procedemos al cálculo del valor presente neto en la siguiente tabla 63.

Tabla 63*Evaluación económica expresada en soles*

<i>Periodo</i>	<i>2024</i>	<i>2025</i>	<i>2026</i>	<i>2027</i>
Inversión	-216,400			
Ahorros Esperados		214,797	193,317	173,985
Mantenimiento y Auditoria		-21,640	-21,640	-21,640
Otros ingresos			44,045	23,938
Flujo Neto		193,157	215,722	176,283
VPN	107,422			
TIR	41.05%			

Nota. Elaboración propia.

Los ingresos esperados como se muestran en la tabla 63, decaen después del primer año a razón de 10% por conceptos de incrementos salariales, políticas de retención e inflación año tras año, lo que merma el ingreso proyectado a esa tasa anual. La tasa de descuento es de 22.6% acorde al costo promedio ponderado del capital para MS4M del año en curso, de acuerdo al costo total del capital y del capital de MS4M conforme se muestra en los anexos D y E, teniendo como costo de deuda 8.92% en promedio y del capital propio en 25%.

El valor presente neto en positivo y la tasa interna de retorno en 41.5% superior a la tasa de descuento de 22.6% indican claramente que la propuesta es viable económicamente. Sin embargo, se realizó el análisis de sensibilidad para la toma de decisiones y evaluar posibles escenarios que afecten el proyecto, como se muestra en la siguiente tabla 64:

Tabla 64*Sensibilidad del factor de escalamiento a nivel 3*

<i>Escalamiento</i>	<i>VAN en soles</i>
20.0%	-S/217,619.05
19.5%	-S/190,532.32
19.0%	-S/163,445.59
18.5%	-S/136,358.86
18.0%	-S/109,272.13
17.5%	-S/82,185.41
17.0%	-S/55,098.68
16.5%	-S/28,011.95
16.0%	-S/925.22
15.5%	S/26,161.51
15.0%	S/53,248.24
14.5%	S/80,334.97
14.0%	S/107,421.69
13.5%	S/134,508.42
13.0%	S/161,595.15
12.5%	S/188,681.88
12.0%	S/215,768.61
11.5%	S/242,855.34
11.0%	S/269,942.06
10.5%	S/297,028.79
10.0%	S/324,115.52

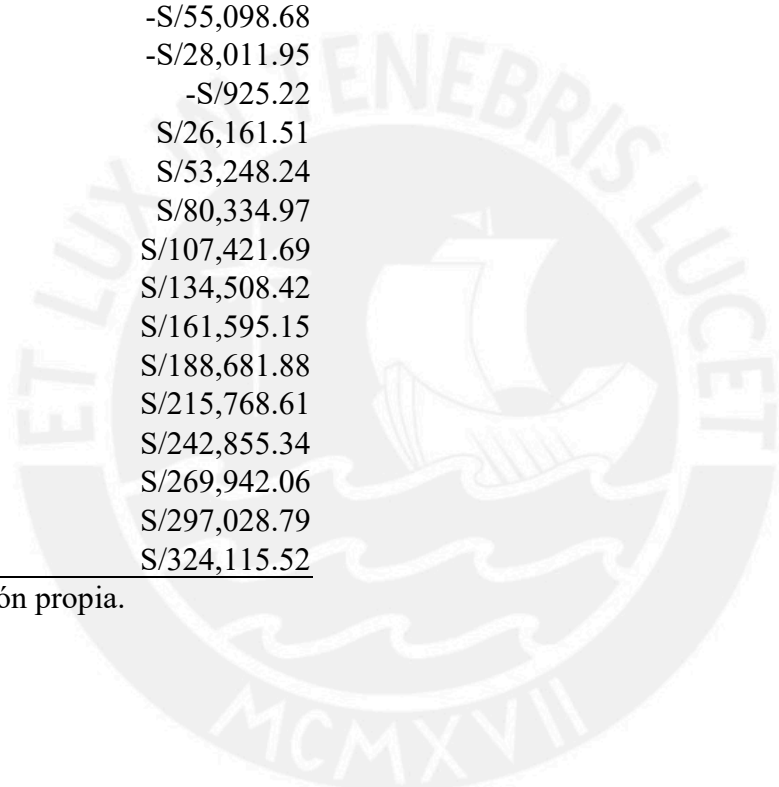
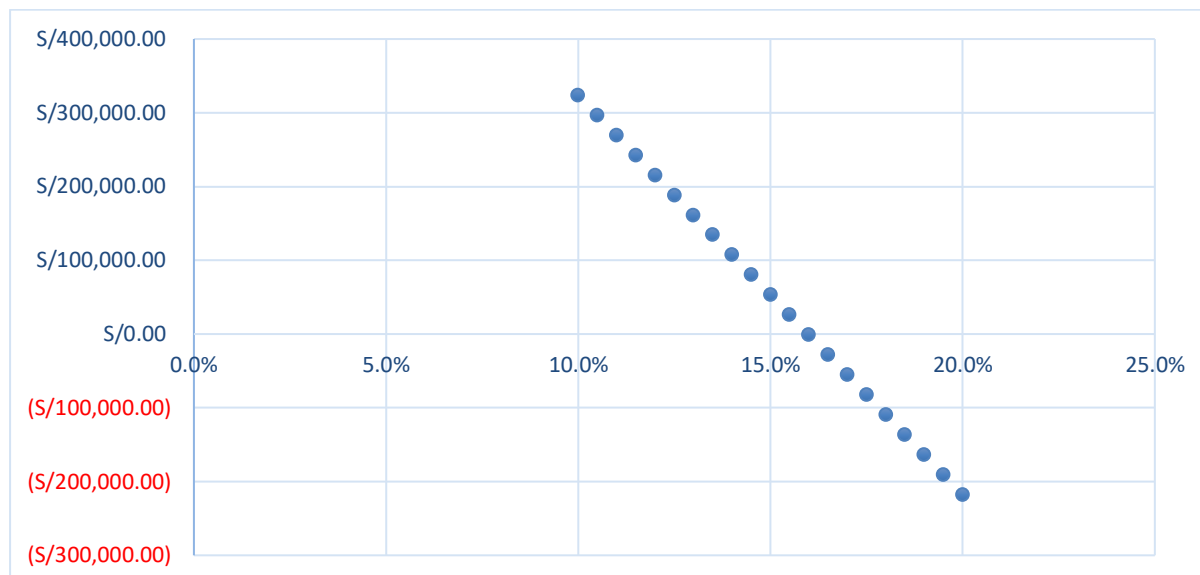
Nota. Elaboración propia.

Figura 33

Sensibilidad del escalamiento proyectado versus el valor actual neto



Nota. Elaboración propia.

Como se observa en la figura 33, existe una fuerte necesidad de búsqueda de la eficiencia en los usos de los recursos para el escalamiento y atención a nivel 3 de los incidentes por parte del personal de investigación y desarrollo. La inversión será factible sólo si se mantiene la disciplina en la implementación y la obtención de la calidad de producto buscado con la implementación de políticas de gobernanza, arquitectura empresarial y practicas ágiles en el menor tiempo.

Otros resultados esperados de acuerdo con los enfoques presentados en la solución propuesta son difíciles de cuantificar en un proceso contable auditable, pero es necesario mencionarlos como beneficios intangibles que se obtendrán en esta implementación.

Fortalecimiento en el desarrollo de software.

La adopción del marco de referencia le permitirá a MS4M la obtención de otros objetivos estratégicos plasmados tales como la certificación de sus procesos de desarrollo de software, la obtención de credenciales de Great Place to Work en un mediano plazo, y penetración en mercados por un mejor desempeño técnico de sus productos.

Mejora en la seguridad y la calidad del software.

Incorporación de prácticas de seguridad desde el inicio del ciclo de vida del desarrollo de software mediante la adopción de DevSecOps, reducirá el número de brechas y parches necesarios, haciendo sus procesos auditables con miras a una posible certificación y homologación con sus clientes, incrementando su confianza y permitiéndole escalar en el mercado.

Capacidad de retención y atracción de talento.

Establecer metodologías estándares en el mercado le permitirá ser más atractivo para otros talentos locales y tendrá una mayor capacidad para retener talento mediante la incorporación de mejores prácticas en la industria.

Cambio cultural y organizacional positivo.

Desarrollo de una cultura organizacional basada en la transparencia, la responsabilidad y la colaboración, lo que promueve un entorno de trabajo más motivador y productivo. Mejora en la comunicación y la colaboración entre equipos y departamentos, lo que facilita la implementación de cambios y la resolución de problemas de manera más efectiva.

9.2. Resultados esperados en el frente social

La implementación de las alternativas de solución identificadas, en adición a los efectos positivos en MS4M, contarán con un efecto también positivo en el desarrollo del

entorno interno y externo de la organización. A continuación, se detallan algunos beneficios estimados.

Desarrollo Profesional

Participar en la implementación y adopción de las alternativas de solución identificadas. Proporcionará al personal local de Cajamarca la oportunidad de trabajar en el proceso de adopción de estándares y prácticas como las metodologías ágiles, ampliará su experiencia y conocimientos en el campo del desarrollo de tecnologías, enriqueciendo su trayectoria profesional.

Contribución al Sector Económico:

La implementación de las alternativas de solución identificada en MS4M, aportará en la calidad del desarrollo de mejorar la eficiencia y calidad en el desarrollo de software, lo que se conllevaría en un mejor desempeño en su mercado y a mostrarse como competidor serio ante empresas de mayores recursos ampliando su ventaja competitiva, pero sobre todo impactando en el desarrollo económico de Cajamarca al fortalecer la competitividad y la capacidad de innovación de las empresas locales en el ámbito tecnológico.

Crecimiento Descentralizado

Al implementar modelos de gobernanza organizacional y arquitectura empresarial, se fomentará un crecimiento descentralizado en MS4M, coadyuvando a que la toma de decisiones y responsabilidades se distribuyan de manera equilibrada entre los equipos, favoreciendo a la creación de un ambiente de trabajo colaborativo para el personal, permitiendo al personal, asumir nuevos roles de liderazgo.

Capítulo X: Conclusiones y recomendaciones

10.1. Conclusiones

MS4M ha acumulado en su historia, una cantidad de deuda técnica considerable, representada en costos de soporte y mantenimiento de sus productos que representan un uso de recursos destinados a la investigación y desarrollo de nuevos productos o evoluciones de estos en el orden de un 20% adicional actualmente, producto de su necesidad de penetración en el mercado, madurez muy baja de sus procesos y por los modelos de negocios adoptados.

Los modelos de negocios actuales de MS4M y su plan estratégico, decantan la necesidad de gestionar la deuda técnica para que pueda conseguir y mantener resultados financieros y de mercado para mantenerse vigente y competitivo. MS4M, así como la penetración de nuevos mercados internacionales que estratégicamente ha planeado en un período de 7 años.

Los problemas relacionados a la deuda técnica de MS4M están vinculados a la calidad de arquitectura y de código, concentrados en el desarrollo de software. La literatura consultada identifica a la adopción de marcos de referencia de metodologías ágiles como DevSecOps como una valiosa herramienta para evitar seguir acumulando una deuda técnica que incremente los costos de desarrollo y que le disminuya competitividad en el mercado.

Sin embargo, la literatura indica que los marcos de referencia requieren de lineamientos de gobierno y de una arquitectura que la sostenga, de forma tal que produzca el impacto que se desea en la implementación.

Esta implementación puede generar ahorros en los costos de soporte directamente, y de esta forma obtener otros beneficios intangibles tales como un incremento en la capacidad de retención de personal clave de investigación y desarrollo, disminución de los costos de desarrollo de productos, un menor tiempo de lanzamiento de productos al mercado, una

mejor capacidad de retención de clientes y menos exposición a penalidades, entre otras, que en suma, pueden representar entre 0.2 y 0.5% de las ventas anualizadas. Otros beneficios intangibles relacionados al aspecto social están vinculados a la ciudad de Cajamarca, donde MS4M desarrolla el software, brindando a los profesionales de esa localidad oportunidad de elevar sus competencias profesionales.

La inversión requerida para la adopción de metodologías ágiles como DevSecOps junto las políticas de gobierno requeridas además de una definición de arquitectura empresarial que la sostenga, es totalmente rentable y genera un retorno positivo para un escenario estimado de tres años de operación. Se espera reducir las incidencias de escalamiento a nivel 3 desde 20% a 10%, teniendo como punto de equilibrio un 11% de escalamiento, de forma tal que puede prescindir de recursos adoptando metodologías ágiles.

10.2. Recomendaciones

Considerando el análisis de la situación actual de MS4M y las conclusiones obtenidas acerca de su deuda técnica y la importancia de adoptar metodologías ágiles como DevSecOps, se recomienda a la organización lo siguiente:

Establecer mecanismos de seguimiento y cumplimiento de los planes de acción para la gestión de la deuda técnica, priorizando las áreas de mayor impacto a nivel económico y complejidad.

Fortalecer la adopción de metodologías ágiles y respaldar su uso a través del cumplimiento de las políticas implementadas, considerando la inversión de recursos para la adopción de DevSecOps.

Identificar factores de medición y métricas para determinar el impacto de lo implementado, con el objetivo de formular acciones de mitigación y seguimiento del uso de recursos.

Fomentar una cultura en MS4M con enfoque en DevSecOps donde las etapas del ciclo de vida estén integradas y compartidas entre los equipos de Desarrollo, Seguridad y Operaciones.



Referencias

- Abenov, T., Franklin-Hensler, M., Grabbert, T., & Larrat, T. (2023, February 14). *Has mining lost its luster? why talent is moving elsewhere and how to bring them back*. McKinsey & Company. Recuperado de <https://www.mckinsey.com/industries/metals-and-mining/our-insights/has-mining-lost-its-luster-why-talent-is-moving-elsewhere-and-how-to-bring-them-back>
- Abidin, N. H. Z., & Samat, P. A. (2021). Requirements identification for distributed agile team communication using high level carotene. *Bulletin of Electrical Engineering and Informatics*, 10(1), pp. 249-257. doi: 10.11591/eei.v10i1.2031
- Afshar, Y. (2022). Towards a practical software engineering curriculum: A data-centric study. *Master's Thesis, University of Calgary, Calgary, AB, Canada*, Recuperado de <https://prism.ucalgary.ca/server/api/core/bitstreams/784b8a42-ec48-4972-a15a-cf2a1d4f8205/content>
- Akbar, M. A., Smolander, K., Mahmood, S., & Alsanad, A. (2022). Toward successful DevSecOps in software development organizations: A decision-making framework. *Information and Software Technology*, 147. doi: 10.1016/j.infsof.2022.106894
- Albonico, M., Đorđević, M., Hamer, E., & Malavolta, I. (2023). Software engineering research on the Robot Operating System: *A systematic mapping study*. *Journal of Systems and Software*, 197, pp. 1-28. doi: 10.1016/j.jss.2022.111574
- Aldaej, A., Nguyen-Duc, A., & Gupta, V. (2023). A Lean Approach of Managing Technical Debt in Agile Software Projects—A Proposal and Empirical Evaluation. *In International*

Conference on Agile Software Development, pp. 67-76. Cham: Springer Nature Switzerland. doi: 10.1007/978-3-031-33976-9_5

Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the benefits of combining devops and agile. *Future Internet 2022*, 14(2), 63. doi: 10.3390/fi14020063

Anjaria, D., & Kulkarni, M. (2022). Effective DevSecOps Implementation: A Systematic Literature Review. *Cardiometry*, (24), pp. 410-417. doi: 10.18137/cardiometry.2022.24.410417

Aratchige R., Manujaya K., Weerasinghe P. (2024). An Overview of Structural Design Patterns in Object-Oriented Software Engineering. *Software Modeling*. doi: 10.13140/RG.2.2.16089.90724.

Aroral, H. K. (2021). Waterfall Process Operations in the Fast-paced World: Project Management Exploratory Analysis. *International Journal of Applied Business and Management Studies*, 6(1), pp. 91-99. Recuperado de: https://www.ijabms.com/wp-content/uploads/2021/05/05_ARORAL_PB.pdf

Azad, N., & Hyrynsalmi, S. (2023). DevOps critical success factors—A systematic literature review. *Information and Software Technology*. doi: 10.1016/j.infsof.2023.107150

Baloch, M. Z., Hussain, S., Afzal, H., Mufti, M. R., & Ahmad, B. (2021). Software developer recommendation in terms of reducing bug tossing length. *In Security, Privacy, and Anonymity in Computation, Communication, and Storage: 13th International Conference, SpaCCS 2020, Nanjing, China, December 18-20, 2020, Proceedings 13*, pp. 396-407. Springer International Publishing. doi: 10.1007/978-3-030-68851-6_29

- Biazotto, J. P., Feitosa, D., Avgeriou, P., & Nakagawa, E. Y. (2023). Technical debt management automation: State of the art and future perspectives. *Information and Software Technology*. doi: 10.1016/j.infsof.2023.107375.
- Blumberg, S., Das, R., Lansing, J., Motsch, N., Münstermann, B., & Patenge, R. (2022). Demystifying digital dark matter: A new standard to tame technical debt. *McKinsey Digital*. Recuperado de: <https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/demystifying-digital-dark-matter-a-new-standard-to-tame-technical-debt>
- Brée, T., & Karger, E. (2022). Challenges in enterprise architecture management: overview and future research. *Journal of Governance and Regulation*, 11(2), 355–367. doi: 10.22495/jgrv11i2siart15
- Buchholz, S., Bechtel, M., & Briggs, B. (2021). Tech Trends 2021. *Deloitte Insights*. Recuperado de: https://www2.deloitte.com/content/dam/insights/articles/6730_TT-Landing-page/DI_2021-Tech-Trends.pdf
- Carver, J. C., Larrucea, X., Serebrenik, A., & Staron, M. (2022). Technical Debt Problems and Concerns. *IEEE Software*, 39(3), pp. 116-119. doi: 10.1109/MS.2021.3133734
- Chen, Liang. Tong, Tony W. Tan, Shaoqin. Han, Nianchen. (2021). Governance and Design of Digital Platforms: A Review and Future Research Directions on a Meta-Organization. *Journal of Management*. 48(1), pp. 147-184. doi: 10.1177/01492063211045023
- Chawla, A., Harryson, M., Mayer, H., & Sinha, M. (2023). The bottom-line benefit of the product operating model. *McKensey Digital*.
- Damyantov, D., Hristov, A., & Varbanov, Z. (2024). Design patterns over SOLID and GRASP principles in real projects. *Mathematics and Education in Mathematics*, 53, pp. 76-84. doi: 10.55630/mem.2024.53.076-084

- Desai, R., & Nisha, T. N. (2021). *Best Practices for Ensuring Security in DevOps: A Case Study Approach in Journal of Physics: Conference Series*, 1964. doi: 10.1088/1742-6596/1964/4/042045
- Dlamini, G., Ergasheva, S., Kholmatova, Z., Kruglov, A., Sadovykh, A., Succi, G., ... & Zouev, E. (2022). Metrics for software process quality assessment in the late phases of SDLC. *In Science and Information Conference*, pp. 639-655. Cham: Springer International Publishing. doi: 10.1007/978-3-031-10461-9_44
- Domínguez, L. A. P., Pérez, E. A., & Castro, E. L. (2021). Aplicación de los métodos AMEF-TOPSIS-AHP para determinar el RPN. *Mundo FESC*. 11(21), pp. 37-46. Recuperado de: <https://www.fesc.edu.co/Revistas/OJS/index.php/mundofesc/article/download/629/637>
- Dudysheva, E., & Skopin, I. (2023, June). Unstable Teams: Management of Remote User Interface Evaluation in Adaptive Software Development. *In 2023 3rd International Conference on Technology Enhanced Learning in Higher Education. (TELE)*, pp. 104-108. IEEE. doi: 10.1109/TELE58910.2023.10184341
- Dursun, M., & Goker, N. (2022). Evaluation of Project Management Methodologies Success Factors Using Fuzzy Cognitive Map Method: Waterfall, Agile, And Lean Six Sigma Cases. *International Journal of Intelligent Systems and Applications in Engineering*, 10(1), pp 35-43. doi: 10.18201/ijisae.2022.265
- Edison, H., Wang, X., & Convoy, K. (2022). Comparing Methods for Large-Scale Agile Software Development: A Systematic Literature Review. *IEEE Transactions on software engineering*, 48(8), 2709-2731. doi: 10.1109/TSE.2021.3069039

- Edris, S., Belderbos, R., & Gilsing, V. (2024). Types of common R&D partners and knowledge leakage to rivals: The role of IP litigation reputation. *Technovation*, 131. doi: 10.1016/j.technovation.2024.102955
- Feng, Z., Hu, Z., Li, G., Zhang, Y., Zhang, X., & Zhang, H. (2022). Improving mine reclamation efficiency for farmland sustainable use: Insights from optimizing mining scheme. *Journal of Cleaner Production*, 379. doi: 10.1016/j.jclepro.2022.134615
- Gaete, J., Villarroel, R., Figueroa, I., Cornide-Reyes, H., & Muñoz, R. (2021). Enfoque de aplicación ágil con Serum, Lean y Kanban. *Ingeniare. Revista chilena de ingeniería*, 29(1). doi: 10.4067/S0718-33052021000100141
- Galvan-Cruz, S., Muñoz, M., & Mejía, J. (2024). Open-Source Tools and Their Coverage to Implement DevOps. *In New Perspectives in Software Engineering*, pp. 17-38. Cham: Springer Nature Switzerland, 1135 (pp. 17-38), doi: 10.1007/978-3-031-50590-4_2
- Gerostathopoulos, I., Vogel, T., Weyns, D., & Lago, P. (2021). How do we evaluate self-adaptive software systems?: A ten-year perspective of SEAMS. *In 2021 International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, pp. 59-70. doi: 10.1109/SEAMS51251.2021.00018
- Govil, N., Sharma, N., Thakral, D., Garg, S., Dixit, R., & Chaoudhary, S. L. (2023). A Proposed Architecture for Quality based Decision Making Approach for Software Re-engineering. *International Conference on Electronics and Sustainable Communication Systems. (ICESC), Coimbatore*. doi: 10.1109/ICESC57686.2023.10193032
- Groombridge, D. (2023). Gartner Top 10 Strategic Technology Trends 2023. Gartner, Recuperado de <https://www.gartner.com/en/articles/gartner-top-10-strategic-technology-trends-for-2023>

- Gupta, A., Poels, G., & Bera, P. (2022). Using conceptual models in agile software development: a possible solution to requirements engineering challenges in agile projects. *IEEE Access*, 10, pp. 119745-119766. doi: 10.1109/ACCESS.2022.3221428
- Gutiérrez-Ríos, S., Morales, I., & Vargas-Lombardo, M. (2021). A study of new decade of agile and lean methodologies for development software. In *2021 4th International Conference on Information and Computer Technologies (ICICT)*, pp. 122-130. doi: 10.1109/ICICT52872.2021.00028
- Hegedűs, P., Kádár, I., Ferenc, R., & Gyimóthy, T. (2018). Empirical evaluation of software maintainability based on a manually validated refactoring dataset. *Information and Software Technology*, 95, pp. 313-327. doi: 10.1016/j.infsof.2017.11.012
- Hicks, C. M., Lee, C. S., & Ramsey, M. (2024). Developer Thriving: four sociocognitive factors that create resilient productivity on software teams. *IEEE Software*. doi:10.1109/MS.2024.3382957
- Holdener, K. (2022). Dependency Inversion Principle: Software Design. *POGIL Activity Clearinghouse*, 3(4). Recuperado de: <https://pac.pogil.org/index.php/pac/article/view/262>
- Hossain, M. I. (2023). Software Development Life Cycle (SDLC) Methodologies for Information Systems Project Management. *International Journal for Multidisciplinary Research (IJFMR)*, 5(5). doi: 10.36948/ijfmr.2023.v05i05.6223
- Iammarino, M., Zampetti, F., Aversano, L., & Di Penta, M. (2021). An empirical study on the co-occurrence between refactoring actions and self-admitted technical debt removal. *Journal of Systems and Software*, 178, 110976. doi: 10.1016/j.jss.2021.110976
- Ibeh, C. V., Awonuga, K. F., Okoli, U. I., Ike, C. U., Ndubuisi, N. L., & Obaigbena, A. (2024). A Review of Agile Methodologies in Product Lifecycle Management: Bridging Theory and

Practice for Enhanced Digital Technology Integration. *Engineering Science & Technology Journal*, 5(2), pp. 448-459. doi: 10.51594/estj.v5i2.805

Ilin, I. V., Levina, A. I., Dubgorn, A. S., & Abran, A. (2021). Investment models for enterprise architecture (Ea) and it architecture projects within the open innovation concept. *Journal of Open Innovation: Technology, Market, and Complexity*. 7(1), 69. doi: 10.3390/joitmc7010069

IT, BUDDHA. (2023). Tech Debt Defeated: Your Real-Life Guide to Practical Strategies for Tackling Technical Debt in IT Projects. *Kindle Edition*.

Jacobides, M. G., Cennamo, C., & Gawer, A. (2024). Externalities and complementarities in platforms and ecosystems: From structural solutions to endogenous failures. *Research Policy*, 53(1). doi: 10.1016/j.respol.2023.104906

Jarzebowicz, A. Weichbroth, P. (2021). A Systematic Literature Review on Implementing Non-functional Requirements in Agile Software Development: Issues and Facilitating Practices. *Lean and Agile Software Development. LASD 2021. Lecture Notes in Business Information Processing*, 408. Springer, Cham, pp. 91-110. doi: 10.1007/978-3-030-67084-9_6

Jarzębowicz, A., & Weichbroth, P. (2021). A qualitative study on non-functional requirements in agile software development. *IEEE Access*, 9, pp. 40458-40475. doi: 10.1109/ACCESS.2021.3064424

Jodlbauer, H., Tripathi, S., Brunner, M., & Bachmann, N. (2022). Stability of cross impact matrices. *Technological Forecasting and Social Change*, 182. doi: 10.1016/j.techfore.2022.121822

- Karunarathne, M. A. W., Wijayanayake, W. M. J. I., & Prasadika, A. P. K. J. (2024, February). DevOps Adoption in Software Development Organizations: A Systematic Literature Review. *In 2024 4th International Conference on Advanced Research in Computing (ICARC)*, pp. 282-287. IEEE. doi: 10.1109/ICARC61713.2024.10499789
- Kashevarova, N. A. (2024). Analysis of the Methods of Intellectual Property Management in Innovation Ecosystems, *Ecological Footprint of the Modern Economy and the Ways to Reduce It, Advances in Science, Technology & Innovation*, Cham: Springer Nature Switzerland, pp. 221-226. doi: 10.1007/978-3-031-49711-7_38
- Khan, M. S., Khan, A. W., Khan, F., Khan, M. A., & Whangbo, T. K. (2022). Critical Challenges to Adopt DevOps Culture in Software Organizations: A Systematic Review, *IEEE Access*, 10, pp. 14339-14349. doi: 10.1109/ACCESS.2022.3145970
- Kim, G. Humble, J. Debois, P. Willis, J. (2021). The DevOps handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations, *IT Revolution*.
- Krey, M., Kabbout, A., Osmani, L., Saliji, A. (2022). Devops adoption: challenges & barriers. *In: 55th Hawaii International Conference on System Sciences (HICSS), University of Hawai'i at Manoa*, pp. 7297–7309. Recuperado de: <https://scholarspace.manoa.hawaii.edu/items/a27d5a9b-4b32-4c17-b871-e7e817ee9c34>
- Kurnia, S., Kotusev, S., Shanks, G., Dilnutt, R., Taylor, P., & Milton, S. K. (2021). Enterprise architecture practice under a magnifying glass: linking artifacts, activities, benefits, and blockers. *Communications of the Association for Information Systems*, 49(1), 34. doi: 10.17705/ICAIS.04936

- Laluf, G. A. (2022). Implementación de estrategia de desarrollo de producto mediante el uso de metodologías ágiles y de nuevas tecnologías. *Universidad Siglo 21*. Recuperado de: <https://repositorio.21.edu.ar/handle/ues21/24390>
- Laplante, P. A., & Kassab, M. (2022). *What Every Engineer Should Know about Software Engineering*. CRC Press. doi: <https://doi.org/10.1201/9781003218647>
- Lenarduzzi, V., Besker, T., Taibi, D., Martini, A., & Fontana, F. A. (2021). A systematic literature review on technical debt prioritization: Strategies, processes, factors, and tools. *Journal of Systems and Software*, 171. doi: 10.1016/j.jss.2020.110827
- Ley que autoriza la disposición de la Compensación por Tiempo de Servicios a fin de cubrir las necesidades económicas causadas por la pandemia del COVID-19 (2021). Lima, Perú: Congreso de la República del Perú.
- Ligozat, A. L., Lefevre, J., Bugeau, A., & Combaz, J. (2022). Unraveling the hidden environmental impacts of AI solutions for environment life cycle assessment of AI solutions. *Sustainability* 2022; 14 (9): 5172. Towards the Sustainability of AI, 121. doi: 10.48550/arXiv.2110.11822
- Lindelöf, P., & Hellberg, R. (2023). Incubation-An evolutionary process. *Technovation*, 124. doi: 10.5585/2023.25118
- Luo, Y., & Assche, A. (2023). The rise of techno-geopolitical uncertainty: Implications of the United States CHIPS and Science Act. *Journal of International Business Studies*, 54(8), pp. 1423-1440. doi: 10.1057/s41267-023-00620-3
- M. Tuape, V. T. Hasheela-Mufeti, A. Kayanda, J. Porras & J. Kasurinen. (2021). Software Engineering in Small Software Companies: Consolidating and Integrating Empirical

Literature Into a Process Tool Adoption Framework. *IEEE Access*, 9, pp. 130366-130388. doi: 10.1109/ACCESS.2021.3113328

Markets and Markets. (2023). DevOps Market by Type (Solutions and Services), Cloud Deployment Model (Public and Private), Organization Size (SMEs and Large Enterprises), Verticals (Telecommunications, IT & ITES, and BFSI) and Region - Global Forecast to 2028, Devops Market. Recuperado de <https://www.marketsandmarkets.com/Market-Reports/devops-market-824.html>

Melgar, A. S., Vértiz, R. I., Osoreo, H. R. S. M. R., Flores, J. A., Orihuela, V. H. Q., & Lozano, R. A. R. (2021). DevOps as a culture of interaction and deployment in an insurance company. *Turkish Journal of Computer and Mathematics Education*, 12 (4), pp. 1701-1708. Recuperado de: <https://www.turcomat.org/index.php/turkbilmat/article/download/1429/1198>

Melo, A., Fagundes, R., Lenarduzzi, V., & Barbosa Santos, W. (2022). Identification and measurement of Requirements Technical Debt in software development: A systematic literature review. *The Journal of Systems & Software*, 194. doi: 10.1016/j.jss.2022.111483

Menzel, L. M. (2021). Investigating the Adoption and Management of Metrics in Large-Scale Agile Software Development at a German IT-Provider (Doctoral dissertation, Technische Universität München). Recuperado de: <https://www.matthes.in.tum.de/file/ccepz7lnxvr2/Sebis-Public-Website/-/Master-s-Thesis-Leon-Menzel/Masters/Thesis/Leon/Menzel.pdf>

- Micro Focus. (2022). *Nationwide Building Society optimises software development with Micro Focus ALM Octane, helping it deliver reliable services for members*. Recuperado de <https://www.microfocus.com/en-gb/media/case-study/nationwide-building-society-cs-a4.pdf>
- Microsoft. (2022, 28 de noviembre) With CI/CD improvements and shorter release times, J. J. Keller & Associates focuses on creating impactful features using Split Software's Azure DevOps integration, *Customer Stories*. Recuperado de: <https://customers.microsoft.com/en-us/story/1576317370253756949-split-software-j-j-keller-customer-success-story-united-states-safety-compliance>
- Microsoft. (2024, 29 de mayo). Vodafone uses Azure DevOps to power agile transformation, *Customer Stories*. Recuperado de: <https://customers.microsoft.com/en-us/story/1615431565907633402-vodafone-telecommunications-azure-en-united-kingdom>
- Minería en Línea. (2023, 11 de noviembre). *Minería 4.0: Tecnologías emergentes para una industria más segura y sostenible*. Recuperado de <https://mineriaenlinea.com/2023/11/mineria-4-0-tecnologias-emergentes-para-una-industria-mas-segura-y-sostenible/>
- Mishra, A., & Alzoubi, Y. I. (2023). Structured software development versus agile software development: a comparative analysis. *International Journal of System Assurance Engineering and Management*. 14, pp. 1504-1522. doi: 10.1007/s13198-023-01958-5
- Morales, R. Soh, Z. Khomh, F. Antoniol, G. Chicano , F. (2017). On the use of developers' context for automatic refactoring of software anti-patterns. *The Journal of Systems and Software*, 128, pp. 236-251. doi: 10.1016/j.jss.2016.05.042

- Muhammad A., Siddique A., Noorulhasan Naveed Q., Saleem U., Abul Hasan M., Shahzad B. (2021). Investigating Crucial Factors of Agile Software Development through Composite Approach. *Intelligent Automation & Soft Computing*, 27(1), pp. 15-34. doi: 10.32604/iasc.2021.014427
- Nazarenko, A., Vishnevskiy, K., Meissner, D., & Daim, T. (2022). Applying digital technologies in technology roadmapping to overcome individual biased assessments. *Technovation*, 110. doi: 10.1016/j.technovation.2021.102364
- Nugroho, A. S. E., Tjhin, V. U., Kosasih, W., & Prabowo, H. (2022). Bibliometric analysis of research trend on Agile IT Governance. *International Journal of Economics, Business and Accounting Research (IJEBAR)*, 6(1), pp. 65-79. Recuperado de: <https://www.jurnal.stie-aas.ac.id/index.php/IJEBAR/article/view/2976>
- Nwaila, G. T., Frimmel, H. E., Zhang, S. E., Bourdeau, J. E., Tolmay, L. C., Durrheim, R. J., & Ghorbani, Y. (2022). The minerals industry in the era of digital transition: An energy-efficient and environmentally conscious approach. *Resources Policy*, 78, doi: 10.1016/j.resourpol.2022.102851
- Páramo Sepúlveda, G. A. (2023). Project Definition Rating Index para proyectos de Business Intelligence. (Doctoral dissertation, Escuela Colombiana de Ingeniería). Recuperado de: <https://repositorio.escuelaing.edu.co/handle/001/2579>
- Petit, Y., & Marnewick, C. (2021). Strategic alignment of information technology initiatives in a scaled agile environment. *Journal of Modern Project Management*, 8(3). Recuperado de: <https://journalmodernpm.com/manuscript/index.php/jmpm/article/view/JMPM02501/415>
- Plant, O. H., van Hillegerberg, J., & Aldea, A. (2022). Rethinking IT governance: Designing a framework for mitigating risk and fostering internal control in a DevOps environment.

International Journal of Accounting Information Systems. 45. doi:

10.1016/j.accinf.2022.100560

Rahman, H. U., Raza, M., Afsar, P., & Khan, H. U. (2021). Empirical investigation of influencing factors regarding offshore outsourcing decision of application maintenance. *IEEE Access*, 9, pp. 58589-58608. doi: 10.1109/ACCESS.2021.3073315

Rajapakse, R. N., Zahedi, M., Babar, M. A., & Shen, H. (2022). Challenges and solutions when adopting DevSecOps: A systematic review. *Information and software technology*, 141. doi: 10.1016/j.infsof.2021.106700

Rajapakse, Roshan N.; Zahedi, Mansooreh; Babar, Ali M.; Shen, Haifeng. (2022). Challenges and solutions when adopting DevSecOps: A systematic review. *Information and Software Technology*, 141. doi: 10.1016/j.infsof.2021.106700

Robillard, M. P. (2022). *Introduction to Software Design with Java*. Springer International Publishing.

Rodríguez, P., Urquhart, C., & Mendes, E. (2020). A theory of value for value-based feature selection in software engineering. *IEEE Transactions on Software Engineering*, 48(2), pp. 466-484. doi: 10.1109/TSE.2020.2989666

Sánchez, G. (2023). Principios SOLID en la Automatización de Pruebas de Software para Interfaces de Usuario Web con Selenium WebDriver y Java. *Revista Politécnica de Aguascalientes*, 2. Recuperado de: <http://revistapolitecnicaaags.upa.edu.mx/wp-content/uploads/2023/12/V2116-.pdf>

Schramm, V. B., Damasceno, A. C., & Schramm, F. (2023). Supporting the choice of the best-fit agile model using fitradeoff. *Pesquisa Operacional*, 43, e264750. Recuperado de <http://revistapolitecnicaaags.upa.edu.mx/wp-content/uploads/2023/12/V2116-.pdf>

- Shastri, Y., Rashina H., & Robert A. (2021). The role of the project manager in agile software development projects, *Journal of Systems and Software*, 173. doi: 10.1016/j.jss.2020.110871
- Smiari, P., Bibi, S., Ampatzoglou, A., & Arvanitou, E. M. (2022). Refactoring embedded software: A study in healthcare domain. *Information and Software Technology*, 143. doi: 10.1016/j.infsof.2021.106760
- Swan, J., Adriaensen, S., Brownlee, A., Hammond, K., Johnson, C., G., Kheiri, A., Krawiec, F., Merelo, J., Minku, L., Özcan, E., Pappa, G., García-Sánchez, P., Sörensen, K., Voß, S., Wagner, M., White, D. (2022). Metaheuristics “in the Large”. *European Journal of Operational Research*, 297. doi: 10.1016/j.ejor.2021.05.042
- Swart, A., Nell, J., Muller, T., Wing, J. (2021, 20 de julio), The future of work in mining What will jobs look like in intelligent mining operations?, *Deloitte Insights*. Recuperado de: <https://www2.deloitte.com/us/en/insights/industry/mining-and-metals/future-of-mining-industry.html>
- Tashtoush, Y. M., Darweesh, D. A., Husari, G., Darwish, O. A., Darwish, Y., Issa, L. B., & Ashqar, H. I. (2021). Agile approaches for cybersecurity systems, IoT and intelligent transportation. *IEEE Access*, 10, pp. 1360-1375. doi: 10.1109/ACCESS.2021.3136861
- Tuape, M., Iiyambo, P., & Kasurinen, J. (2022). Organizational governance: Resolving insufficient practice and quality expectation in Small Software Companies. In *Proceedings of the 2022 European Symposium on Software Engineering*, pp. 17-24. doi: 10.1145/3571697.3571700

- Uysal, Murat Pasa. Mergen, A. Erhan. (2021). Smart manufacturing in intelligent digital mesh: Integration of enterprise architecture and software product line engineering. *Journal of Industrial Information Integration*, 22, doi: 10.1016/j.jii.2021.100202
- Vogel-Heuser, B., & Bi, F. (2021). Interdisciplinary effects of technical debt in companies with mechatronic products—a qualitative study. *Journal of Systems and Software*, 171. doi: 10.1016/j.jss.2020.110809
- Yang, S., Moon, J., Kim, J., Lee, K., & Lee, K. (2023). FLScalize: Federated learning lifecycle management platform. *IEEE Access*, 11. doi: 10.1109/ACCESS.2023.3275439
- Zielske, M., & Held, T. (2021). Application of agile methods in traditional logistics companies and logistics startups: Results from a German Delphi Study. *The Journal of Systems and Software*. 177. doi: 10.1016/j.jss.2021.110950

Equipo de Investigación. (2023). *Encuesta sobre la Percepción de causas de alta demanda de soporte técnico de los productos de MS4M.*



Anexo A: Ficha Técnica encuestas

Título de la encuesta:

Encuesta de evaluación de problemática de desarrollo

Fecha de inicio y fin:

24 agosto 2023 al 25 agosto 2023

Objetivo de la encuesta:

Determinación de causa de problemas en el proceso de desarrollo.

Responsable de la encuesta:

Richar Balboa

Población objetivo:

Personal del equipo de desarrollo

Tamaño de muestra:

45 personas

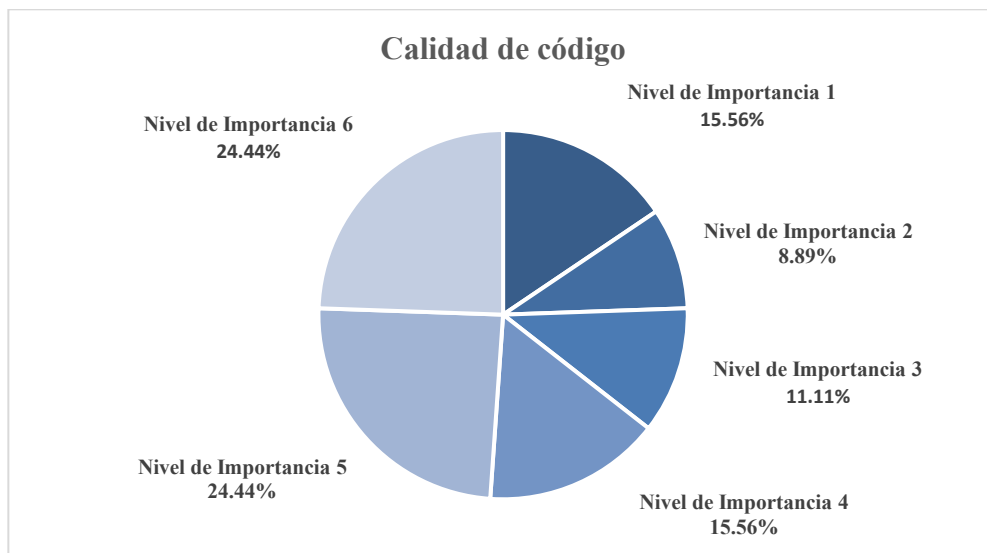
Instrumento de recopilación:

Cuestionario en línea.

A continuación, se muestran los resultados detallados de las encuestas realizadas:

Figura 34

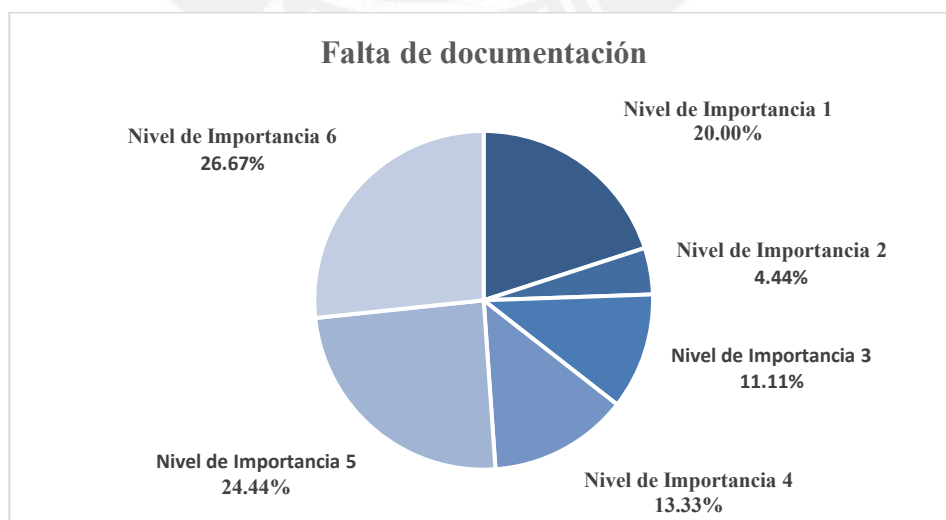
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Calidad de código.



Nota. Elaboración propia

Figura 35

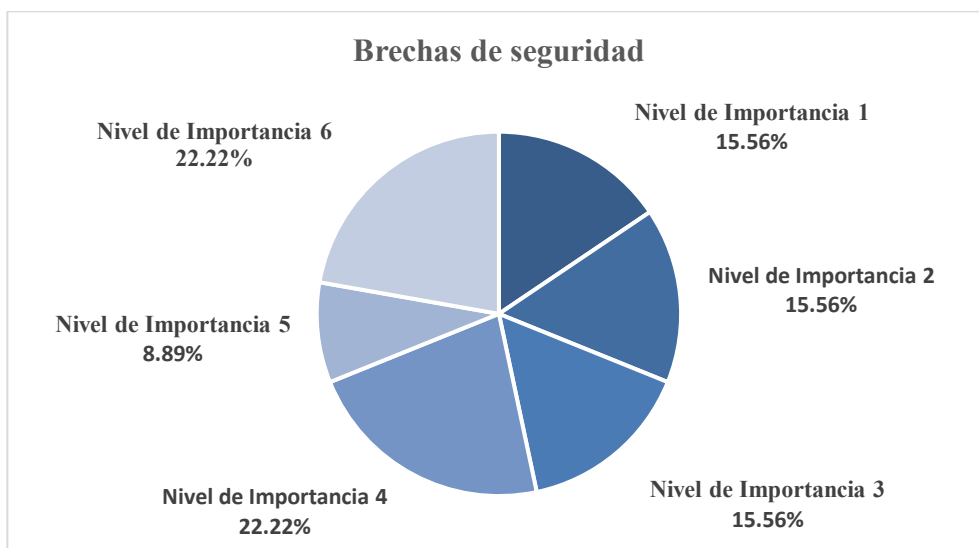
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Falta de documentación



Nota. Elaboración propia

Figura 36

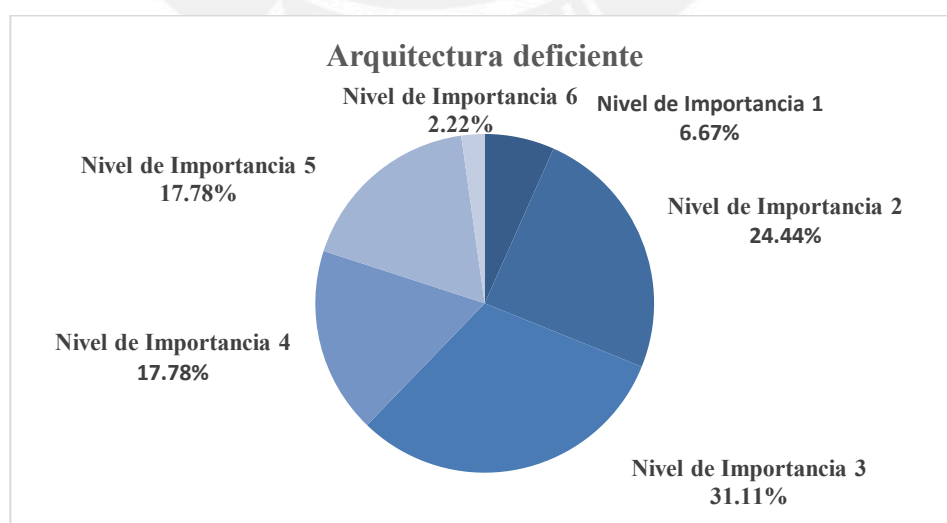
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a las Brechas de Seguridad



Nota. Elaboración propia

Figura 37

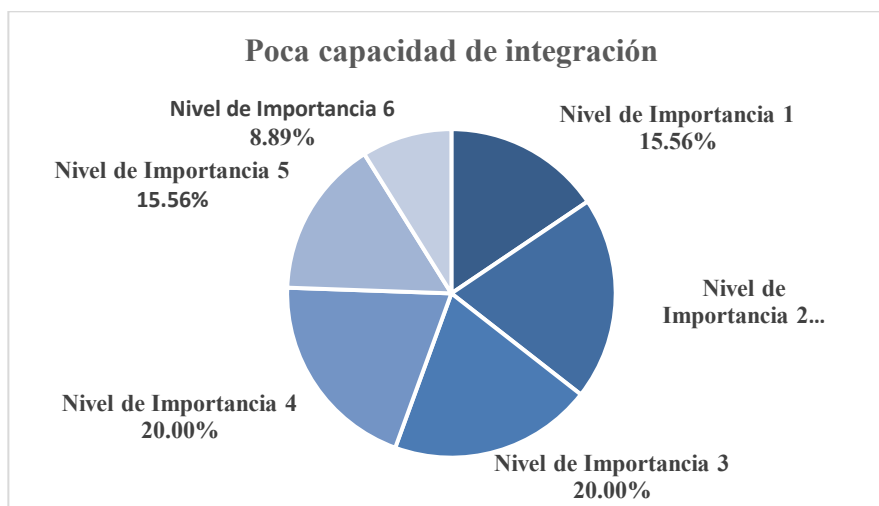
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a la Arquitectura Deficiente



Nota. Elaboración propia

Figura 38

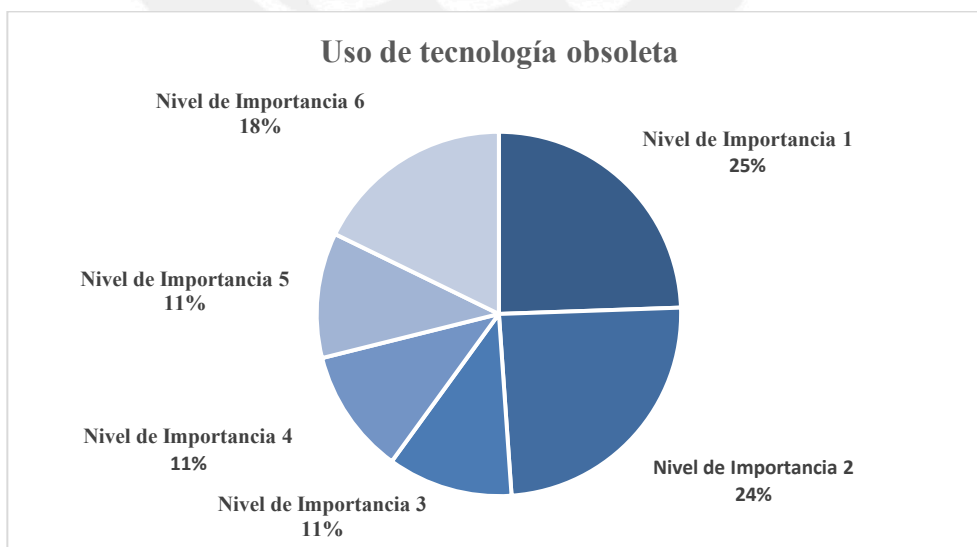
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto a Poca capacidad de Integración



Nota. Elaboración propia

Figura 39

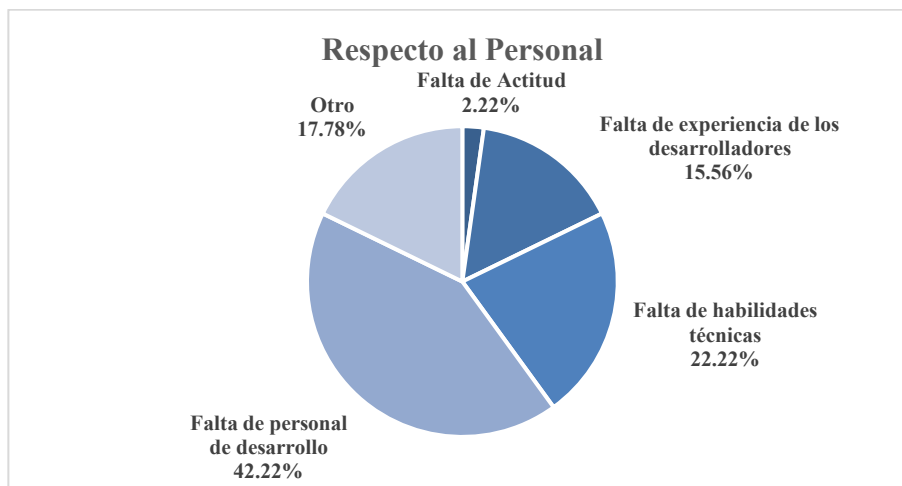
Diagrama de escala de importancia o razón sobre el origen de problemas de alta incidencia respecto al Uso de Tecnología Obsoleta



Nota. Elaboración propia

Figura 40

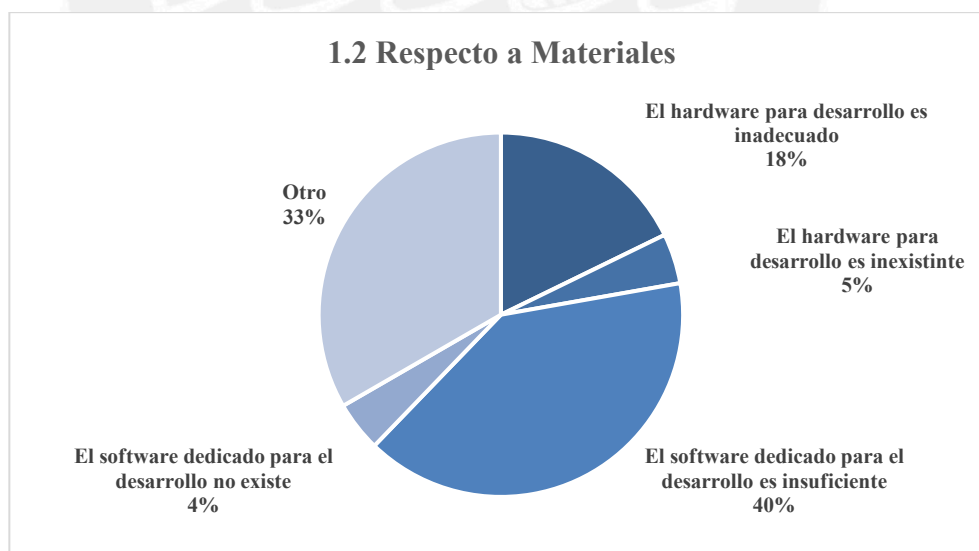
Valoración del aspecto “Personal” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 41

Valoración del aspecto “Materiales” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 42

Valoración del aspecto “Indicadores” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 43

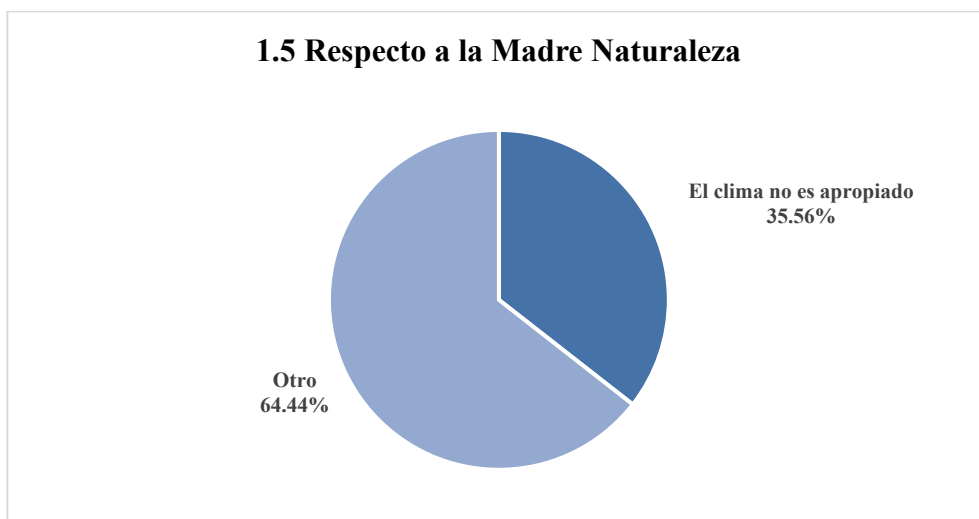
Valoración del aspecto “Mantenimiento” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 44

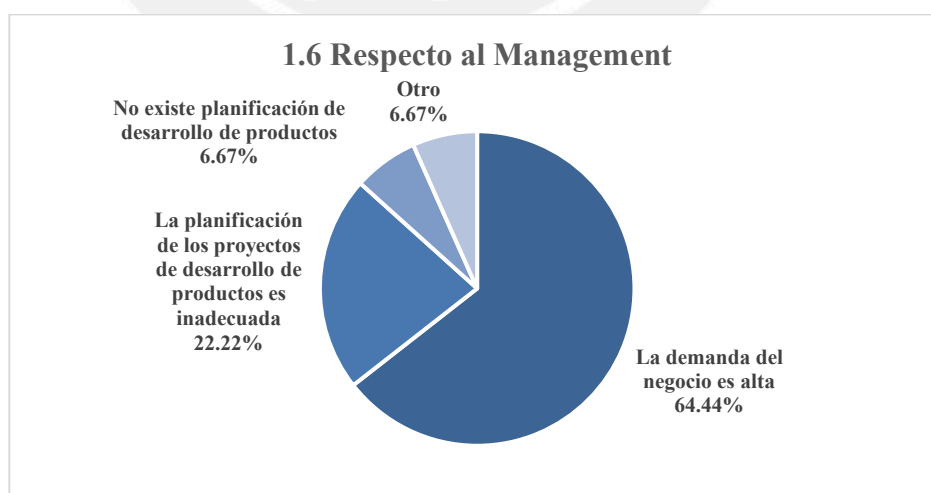
Valoración del aspecto “Madre Naturales” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 45

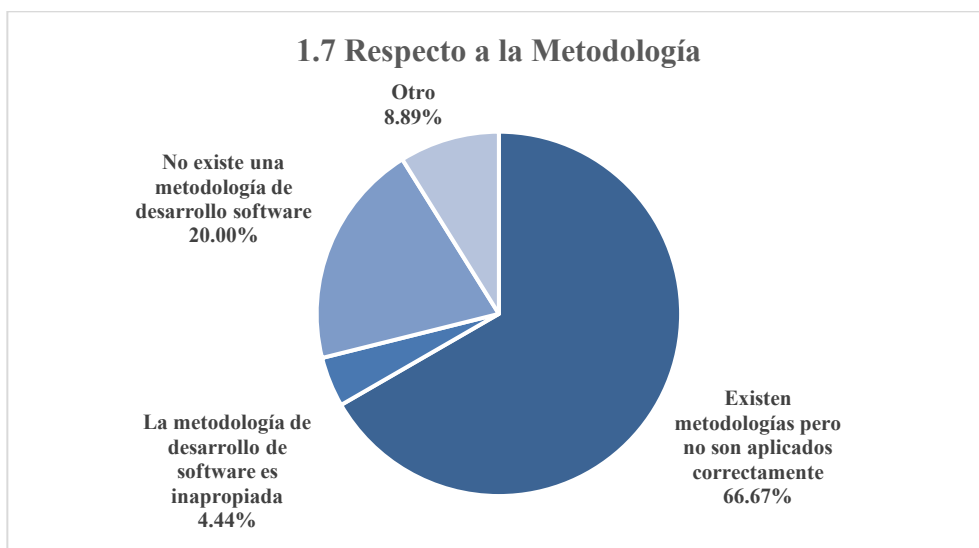
Valoración del aspecto al “Management” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 46

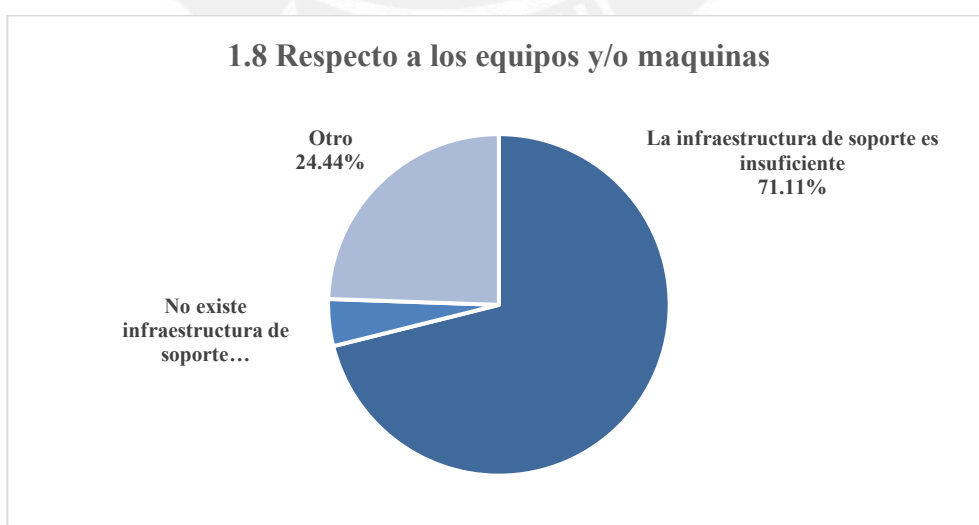
Valoración del aspecto “Respecto a la Metodología” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Figura 47

Valoración del aspecto “Equipos y Maquinas” sobre los tickets Nivel 3 en la atención de incidencias



Nota. Elaboración propia

Anexo B: Objetivos de corto plazo y tablero de mando de MS4M – Objetivo de largo plazo relacionado a las ventas

Short-term Objectives and Balance Score Card



	Objetivos	Responsables	Descripción del Indicador	Unidad de Medida	2021 (Línea Base)	2022	2023	2024	2025	2026	2027	BSC Perspective
OCP1	MS4M al 2027 tendrá Ingresos por ventas superiores a 12.5 millones de USD al año	Gerencia General	ventas	millones de dolares	6.5	6	8	8.5	10	11.5	12.5	
OCP1.1	Ingresar al segmento de minería no metálica con solución para flota ligera	Gerencia General	Ventas relacionadas al segmento	millones de dolares/año	0	0	0.5	0.5	1	1	1.5	Financial
OCP1.2	Ingresar al segmento de mercado de consultoría de analítica de datos para las áreas operacionales	Gerencia General	Ventas relacionadas al segmento	millones de dolares/año		0.1	0.15	0.2	0.2	0.2	0.3	Financial
OCP1.3	Establecer área de gestión comercial y marketing	Gerencia General	Area comercial y de mkt	Cantidad			1					Internal Process
OCP1.4	Ingresar en UG, desarrollar el producto a partir de las primeras operaciones	Gerencia General	Ventas	millones de dolares/año	0	0.3	0.8	1	2	3	3	Financial
OCP1.5	Establecer área de innovación para la creación de nuevos productos.	Gerencia General	Area de investigación implementada	Cantidad				1				Internal Process



Anexo C: Objetivos de corto plazo y tablero de mando de MS4M – Objetivo de largo plazo relacionado a la rentabilidad

Short-term Objectives and Balance Score Card



	Objetivos	Responsables	Descripción del Indicador	Unidad de Medida	2021 (Línea Base)	2022	2023	2024	2025	2026	2027	BSC Perspective
OCP2	MS4M al 2027 generará un EBITDA de 40%	Gerencia General	EBITDA respecto a las ventas	Porcentaje	20%	20%	25%	30%	30%	35%	40%	
OCP2.1	Reducir los niveles de gastos a 37% del nivel de ventas al 2027	Gerencia General	Gastos respecto a las ventas	Porcentaje	47%	47%	45%	43%	40%	38%	37%	Financial
OCP2.2	Presupuestar semestralmente las ventas del 90% de las iniciativas comerciales a partir del 2022	Gerente General	Iniciativas comerciales presupuestadas versus las ejecutadas	Porcentaje		90%	90%	90%	90%	90%	90%	Financial
OCP2.3	Reducir las existencias de 19% a 10% del 2022	Gerente General	Inventario Valorizado respecto a las ventas	Porcentaje	19%	17%	15%	13%	12%	11%	10%	Financial
OCP2.4	Obtener ahorros en costos de compra, almacenamiento y distribución que representen el 3% de las ventas al 2027 (acumulado)	Gerente General	Ahorros en costos sobre ventas	Porcentaje	0%	0.50%	0.80%	1.40%	1.80%	2.20%	3%	Financial
OCP2.5	Optimización del Headcount, lo cual permitirá un ahorro en gastos de 2 millones al 2027	Gerente General	Valor de ahorros en gastos	millones de dolares	0	0.3	0.6	0.9	1.4	1.7	2	Internal Process
OCP2.6	Rediseñar la ControlScreen para hacerlo plug and play y de bajo costo al 2024	Gerente General	Diseño nuevo	Cantidad				1				Internal Process



Anexo D: Estimación de costo promedio ponderado del capital en MS4M



MS4M BALANCE SHEET TO 31/12/2023 [USD]*									
Assets			Liabilities						
	2024	%	2023	%					
Cash and cash equivalents	\$ 179,969	2%	\$ 145,668	2%	Taxes payable	\$ 35,924	0%	\$ 94,768	1%
Trade Receivables	\$ 1,423,867	18%	\$ 1,884,856	26%	Wages payable	\$ 503,897	6%	\$ 271,953	4%
Receivables due from Shareholders, Directors and Staff	\$ 18,882	0%	\$ 6,497	0%	Trade payables	\$ 477,417	6%	\$ 291,741	4%
Other receivables	\$ 61,281	1%	\$ 105,459	1%	Payables to third parties	\$ 98,429	1%	\$ 100,000	1%
Inventories	\$ 1,226,409	16%	\$ 868,694	12%	Financial obligations	\$ 70,296	1%	\$ 179,563	2%
Income tax credit	\$ 188,799	2%	\$ 228,844	3%	Other Payables	\$ 171,545	2%	\$ 171,282	2%
Prepaid expenses	\$ 218,188	3%	\$ 67,521	1%	Current Liabilities	\$ 1,357,508	17%	\$ 1,109,308	15%
Current assets	\$ 3,314,395	42%	\$ 3,307,539	45%	Financial obligations	\$ 756,344	10%	\$ 260,671	4%
Equity Investment	\$ 45,831	1%	\$ 8,343	0%	Deferred Income	\$ -	0%	\$ -	0%
Right of use assets net	\$ 45,658	1%	\$ 55,784	1%	Non Current Liabilities	\$ 756,344	10%	\$ 260,671	4%
Property, plant and equipment net	\$ 1,040,929	13%	\$ 883,861	12%	Total liabilities	\$ 2,113,852	27%	\$ 1,369,979	19%
Intangible assets	\$ 3,358,599	43%	\$ 2,985,826	41%	Share Capital	\$ 161,196	2%	\$ 161,196	2%
Deferred Assets	\$ 77,013	1%	\$ 47,170	1%	Retained earnings	\$ 5,379,241	68%	\$ 4,457,002	61%
Non Current assets	\$ 4,568,029	58%	\$ 3,980,985	55%	Current year result	\$ 228,135	3%	\$ 1,300,348	18%
Total assets	\$ 7,882,424		\$ 7,288,523	100%	Total equity	\$ 5,768,571	73%	\$ 5,918,544	81%
					Total liabilities and equity	\$ 7,882,424	108%	\$ 7,288,523	100%

	Q2-2024	Q2-2024
Capital Propio	361,196	5,540,437
Deuda	826,640	826,640
Costo de Capital Propio	25.00%	25.00%
Costo de Deuda	8.92%	8.92%
Tasa Impositiva	29.50%	29.50%
Value	987,836	6,367,076
EQUITY	0.04	0.22
DEBT	0.05	0.01
WACC	9.34%	22.57%

Weighted average cost of capital formula

$$WACC = \left(\frac{E}{V} \times Re \right) + \left(\frac{D}{V} \times Rd \times (1 - Tc) \right)$$

- E = market value of the firm's equity
- D = market value of the firm's debt
- Tc = corporate tax rate
- Re = cost of equity
- Rd = cost of debt
- V = E + D



**Anexo E: Cálculo de Costo de Deuda en Nota 19a de Estado Financiero al 2023
de MS4M S.A.C.**



FINANCIAL OBLIGATIONS Note 19a

Cuenta	CUENTAS POR PAGAR DIVERSAS A TERCEROS	DESCRIPCION			PEN	USD
452101	BANCO DE CREDITO DEL PERU	LEASING	Corriente	8.0%	96,984.27	25,709.88
451101	BANCO DE CREDITO DEL PERU	CAPITAL DE TRABAJO 2024	Corriente	8.5%	482,800.35	127,987.12
451101	BANCO DE CREDITO DEL PERU	10-Agosto-2018	Corriente	11.0%	28,035.05	7,431.90
					607,819.67	161,128.90
451101	BANCO DE CREDITO DEL PERU	CREDITO HIPOTECARIO	No Corriente	9.0%	688,722.51	182,575.70
451101	BANCO DE CREDITO DEL PERU	MYPERU	No Corriente	8.0%	1,653,566.32	438,349.29
455101	COSTO DE FINANCIACION	CREDITO HIPOTECARIO	No Corriente	9.0%	168,189.84	44,585.99
					2,510,478.67	665,510.97
		Promedio		8.9%		
					3,118,298.34	826,639.87

