Instituto Tecnológico de Costa Rica
Área Académica de Gerencia de Proyectos
Maestría en Gerencia de Proyectos

ACTA DE APROBACION DEL TRABAJO FINAL DE GRADUACIÓN

Con fundamento en lo establecido por la Maestría en Gerencia de Proyectos, el Tribunal Examinador del Trabajo Final de Graduación denominado:

Plan de gestión para el proyecto Downtime System

Habiendo analizado el resultado del trabajo presentado por la estudiante:

Primer Apellido	Segundo Apellido	Nombre	No. de carné
BOLAÑOS	SALAZAR	MARIELA	2023800298

Emite el siguiente dictamen:

● APROBADO	○ REPROBADO ○ SE RECOMIENDA ○ NO SE RECOMIENDA Brindarle una nueva oportunidad para la DEFENSA PÚBLICA de su proyecto final. NUEVA FECHA:_____

Dando fé de lo aquí expuesto firmamos:

LUDBYG JOHANES NOVO GUTIERREZ (FIRMA)
Firmado digitalmente por LUDBYG JOHANES NOVO GUTIERREZ (FIRMA)
Fecha: 2026.01.19 22:47:55 -06'00'

Ing. Ludbyg Novo Gutiérrez, MGP, MBA
Profesor Guía

DONALD ANDRES MUÑOZ CASTRO (FIRMA)
Firmado digitalmente por DONALD ANDRES MUÑOZ CASTRO (FIRMA)
Fecha: 2026.01.21 05:32:06 -06'00'

Ing. Donald Muñoz Castro, MGP
Profesor Lector

ENRIQUE ALONSO HERNANDEZ GOMEZ (FIRMA)
Firmado digitalmente por ENRIQUE ALONSO HERNANDEZ GOMEZ (FIRMA)
Fecha: 2026.01.27 09:27:33 -06'00'

Ing. Alonso Hernández Gómez, MG
Profesor Lector

MILTON ANTONIO SANDOVAL QUIROS (FIRMA)
Firmado digitalmente por MILTON ANTONIO SANDOVAL QUIROS (FIRMA)
Fecha: 2026.01.16 13:45:56 -06'00'

Ing. Milton Sandoval Quirós., MBA
Maestría en Gerencia de Proyectos

Jueves 15 Enero 2026

INSTITUTO TECNOLÓGICO DE COSTA RICA
ÁREA ACADÉMICA DE GERENCIA DE PROYECTOS
MAESTRÍA EN GERENCIA DE PROYECTOS

Plan de gestión para el proyecto Downtime System

Proyecto Final de Graduación para optar por el título de
Máster en Gerencia de Proyectos en el énfasis de proyectos de TI
con el grado académico de Maestría

Realizado por:
Mariela Bolaños Salazar

Cartago, octubre, 2025

DEDICATORIA

Dedico este trabajo a mi familia, por su apoyo incondicional y constante motivación durante todo el proceso académico y profesional. A mis padres, por enseñarme el valor del esfuerzo y la perseverancia, y a quienes han creído en mi capacidad para superar cada reto. Este logro es tan suyo como mío.

AGRADECIMIENTOS

Agradezco profundamente al Instituto Tecnológico de Costa Rica y al Área Académica de Gerencia de Proyectos por brindarme las herramientas y conocimientos necesarios para el desarrollo de este proyecto. Mi gratitud a los profesores y asesores que guiaron mi formación y aportaron valiosas recomendaciones. Extiendo mi agradecimiento al equipo de Tec Ops de Boston Scientific Costa Rica, especialmente a los colaboradores de los departamentos de Materiales, Equipos e Industrial, por su disposición y colaboración en la recopilación de información y validación de requerimientos. Finalmente, agradezco a mis compañeros y amigos por su apoyo y consejos durante este proceso.

EPÍGRAFE

"El éxito de un proyecto no depende solo de la tecnología, sino de la capacidad de las personas para adaptarse, colaborar y aprender continuamente." Project Management Institute, Guía del PMBOK® 7ma edición (2021)

ÍNDICE GENERAL

DEDICATORIA ..... *i*

AGRADECIMIENTOS ..... *ii*

EPÍGRAFE ..... *iii*

ÍNDICE GENERAL ..... *iv*

ÍNDICE DE FIGURAS ..... *viii*

ÍNDICE DE TABLAS ..... *x*

ÍNDICE DE CUADROS ..... *xi*

LISTA DE ABREVIATURAS ..... *xiii*

RESUMEN ..... *xiv*

ABSTRACT ..... *xv*

INTRODUCCIÓN *1*

Capítulo 1 Generalidades de la investigación *3*

1.1 Marco de referencia organizacional **3**

1.1.1 Reseña histórica Boston Scientific. 3

1.1.2 Estructura y marco estratégico. 4

1.1.3 Proyectos en la organización. 7

1.1.4 Proyecto Downtime System 12

1.2 El problema y su impacto **15**

1.3 Objetivos **20**

1.3.1 Objetivo general. 20

1.3.2 Objetivos específicos. 20

1.4 Alcance y limitaciones **20**

1.4.1 Alcance 20

1.4.2 Limitaciones. 21

Capítulo 2 Marco teórico *22*

2.1 Proyectos de Software y su gestión **22**

2.1.1 Definición de proyecto. 22
2.1.2 Proyectos de software. 23
2.1.3 Gestión de los proyectos. 24
2.2 Practicas actuales en desarrollo de Software 31
2.3 Guía de buenas prácticas estándar PMBOK 32
2.4 Agilismo y sus marcos 34
2.4.1 Scrum 35
2.4.2 Kanban 39
2.5 Herramientas tecnológicas para gestión de proyectos. 40
2.5.1 Características clave de las herramientas de gestión de proyectos. 42
2.5.2 Las mejores 5 herramientas de gestión de proyectos. 43
Capítulo 3 Marco Metodológico 46
3.1 Categorías de la investigación 47
3.2 Población y muestra – Sujetos de investigación 49
3.3 Fuentes de información 50
3.4 Técnicas y herramientas para la recopilación de datos 51
3.4.1 Revisión bibliográfica y Juicio experto 51
3.4.2 Focus group 52
3.4.3 Entrevista semiestructurada 54
3.4.4 Filtros de idoneidad y Juicio experto 54
3.4.5 Entrevista abierta 55
3.5 Procesamiento y productos de la investigación 56
3.5.1 Productos de la investigación 58
3.5.2 Técnicas de procesamiento 61
Capítulo 4 Análisis de Resultados 64
4.1 Análisis de requisitos y generalidades del proyecto 64
4.2 Análisis del repositorio de proyectos anteriores 74
4.3 Análisis de ciclos de vida 77
4.4 Análisis de marco de trabajo 80

4.5 Análisis final ... **87**

Capítulo 5 Propuesta de Solución ... ***88***

5.1 Ciclo de vida a seleccionar ... **88**

5.2 Estructura del plan de gestión del proyecto ... **89**

5.2.1 Gestión de integración ... 89

5.2.2 Gestión de alcance ... 102

5.2.3 Gestión de cronograma ... 119

5.2.4 Gestión de costos ... 130

5.2.5 Gestión de calidad ... 133

5.2.6 Gestión de interesados y comunicaciones ... 142

5.2.7 Gestión de riesgos ... 151

5.2.8 Gestión de cambios ... 164

5.2.9 Cierre del proyecto ... 166

5.3 Forma de trabajar del equipo ... **168**

5.3.1 Dinámica de trabajo ... 170

Capítulo 6 Conclusiones y Recomendaciones ... ***171***

6.1 Conclusiones ... **171**

6.2 Recomendaciones ... **171**

Capítulo 7 Referencias Bibliográficas ... ***173***

Capítulo 8 Apéndices ... ***176***

Apéndice A: Revisión bibliográfica ... **176**

Apéndice B: Entrevista semiestructurada ... **176**

Apéndice C: Entrevista abierta ... **177**

Apéndice D: Talleres de focus group ... **177**

Capítulo 9 Anexos ... ***179***

Anexo 1: Elección ciclo de vida ... **179**

Anexo 2: Resultado Entrevistas ... **182**

Anexo 2: Talleres focus group ... **185**

Anexo 3: Filtros de idoneidad 185

Anexo 4: Revisión bibliográfica 188

Anexo 5: Seguimiento proyecto Downtime System. Portafoloio Transformación Digital Heredia. 190

ÍNDICE DE FIGURAS

Figura 1.1. *Distribución de las sedes global* 4

Figura 1.2. *Organigrama empresarial* 5

Figura 1.3. *Panel DE Heredia* 12

Figura 1.4. *Procesos en Jira Downtime* 13

Figura 1.5. *Requerimientos en definición Downtime system* 15

Figura 1.6. *Plan Downtime* 16

Figura 1.7. *Interesados Downtime* 17

Figura 1.8. *Documentos Downtime* 18

Figura 1.9. *Ingreso Manual Line Disruption* 19

Figura 2.1. *Ejemplo de ciclo de vida predictivo* 28

Figura 2.2. *Ciclo de vida iterativo* 29

Figura 2.3. *Ciclo vida incremental* 30

Figura 2.4. *Ciclo vida ágil* 31

Figura 2.5. *Fases y procesos Scrum* 37

Figura 2.6. *Tablero Kanban* 40

Figura 2.7. *Características claves en las herramientas de gestión de proyectos* 42

Figura 2.8. *Panel Oddo* 43

Figura 2.9. *Panel Asana* 44

Figura 2.10. *Panel Monday* 44

Figura 2.11. *Panel Trello* 45

Figura 2.12. *Panel Jira* 46

Figura 3.1. *Etapas investigación* 57

Figura 4.1. *Gráfico radar de filtros de idoneidad* 77

Figura 5.1. *Ciclo de vida ágil Downtime* 88

Figura 5.2. *Backlog inicial* 110

Figura 5.3. *Ejemplo Sprint#1 en ejecución* 113

Figura 5.4. *Downtime product backlog* *114*

Figura 5.5*. Sprint #1 Downtime* *114*

Figura 5.6. *Cronograma de desarrollo* *126*

Figura 5.7. *Épica "Mejoras"* *165*

Figura 5.8. *Tec Ops creación de incidente* *168*

ÍNDICE DE TABLAS

***Tabla 1.1**. Aristas Tec Ops* 6

***Tabla 1.2.** Personal con conocimiento en gestión de proyectos* 17

***Tabla 1.3**. Incidentes mes de junio 2025* 19

***Tabla 5.1.** Downtime story points* 127

ÍNDICE DE CUADROS

Cuadro 1.1 *Proyectos Tec Ops* 9

Cuadro 2.1. *Características en los ciclos de vida* 26

Cuadro 3.1. *Categorías de la investigación* 48

Cuadro 3.2. *Sujetos de estudio* 49

Cuadro 3.3. *Fuentes de información primarias* 50

Cuadro 3.4. *Fuentes de información secundarias* 51

Cuadro 3.5. *Contenido temático* 61

Cuadro 3.6. *Conceptos claves* 62

Cuadro 3.7. *Categorías* 62

Cuadro 3.8. *Codificando respuestas* 62

Cuadro 3.9. *Comparando marcos de trabajo* 63

Cuadro 3.10. *Aspectos del proyecto según idoneidad* 63

Cuadro 3.11. *Conociendo el departamento* 63

Cuadro 4.1. *Caracterización del proyecto* 66

Cuadro 4.2. *Análisis del repositorio de proyecto anteriores* 75

Cuadro 4.3. *Análisis del ciclo de vida del proyecto* 78

Cuadro 4.4. *Artefactos y herramientas para gestionar el proyecto* 81

Cuadro 5.1. *Herramientas tecnológicas recomendadas* 96

Cuadro 5.2. *Métricas e indicadores de control del proyecto* 97

Cuadro 5.3. *Métricas gestión alcance* 116

Cuadro 5.4. *Plantilla de Seguimiento y Control* 117

Cuadro 5.5. *Distribución de Sprint y story points* 121

Cuadro 5.6. *Tareas y estimación* 122

Cuadro 5.7. *Fechas claves Downtime* 128

Cuadro 5.8. *Control de cronograma* 130

Cuadro 5.9. *Estimación costo por sprint y recursos* 132

Cuadro 5.10. *Control de costos* *133*

Cuadro 5.11*. Criterios calidad Downtime generales* *135*

Cuadro 5.12. *Métodos para validar calidad de software* *136*

Cuadro 5.13. *Control de revisión sprints* *139*

Cuadro 5.14. *Plantilla Sprint Retrospective* *141*

Cuadro 5.15. *Control de calidad* *141*

Cuadro 5.16. *Matriz de interesados* *143*

Cuadro 5.17. *Reuniones del proyecto* *148*

Cuadro 5.18. *Herramientas de comunicación* *150*

Cuadro 5.19. *Escala de probabilidad e impacto* *154*

Cuadro 5.20. *Mapas de calor* *155*

Cuadro 5.21. *Risk breakdown structure* *157*

Cuadro 5.22. *Control y seguimiento de riesgos* *159*

Cuadro 5.23. *Plantilla informe cierre Downtime* *167*

LISTA DE ABREVIATURAS

BSC: *Boston Scientific*

DE: *Development Excellent*

DevOps: Desarrollo de software y las operaciones de TI

DoD: Definición del *Done*

EHS*: Environment, Health and Safety Department*

GSM: *Global System Methodology*

IA: Inteligencia Artificial

IPM*: In-Process Monitoring*

ISO: Normas mundiales para bienes y servicios de confianza.

MES: *Manufacturing Execution System*

MTTR: *Medt time to Report*

MTBF: *Medt time between fail*

MVP: Producto mínimo viable

PFG: Proyecto Final de Gradución

PM: *Project Manager*

PMBoK®: *Project Management Body of Knowledge.*

PMI®: *Project Management Institute.*

PTS: *Production Tracking System*

PO: *Product Owner*

QA: *Quality Assurance*

RBA: *Risk Breakdown Structure*

SM: Scrum Master

SP: *Story Point*

TI: Tecnologías de la información.

Tec Ops: *Technology for Operations*

RESUMEN

El proyecto *"Downtime System"* se ejecutó en la planta de Heredia de Boston Scientific Costa Rica como respuesta a la necesidad de integrar y optimizar los sistemas de reporte de eventos de líneas detenidas, alineándose con la estrategia de Transformación Digital 2025. El objetivo principal fue proponer y desarrollar un plan de gestión de proyectos que, aplicando buenas prácticas, incrementara las probabilidades de éxito en la implementación del sistema, centralizando la información y mejorando la toma de decisiones operativas.

La gestión del proyecto se fundamentó en un enfoque ágil utilizando el marco de trabajo de Scrum como base, pero adaptando el mismo de acuerdo con las necesidades del proyecto y organización (un toque personal). Esto permitió entregas incrementales, colaboración continua y adaptación a cambios frecuentes en los requerimientos. Se emplearon técnicas como talleres de focus *group*, entrevistas semiestructuradas y revisión bibliográfica para levantar requisitos, identificar buenas prácticas y definir el ciclo de vida más adecuado. El plan de gestión abarcó las áreas clave de integración, alcance, cronograma, costos, calidad, riesgos, comunicaciones y gestión de cambios, apoyándose en herramientas colaborativas como Excel, Jira y Miro.

Entre los principales resultados, se logró caracterizar el proyecto, definir un *backlog* priorizado y establecer métricas de seguimiento como las historias de usuario, el *velocity* del equipo, el *burndown chart* y el tablero de control de calidad. Se identificaron como causas principales de la oportunidad del proyecto de graduación asociados a la falta de experiencia en gestión de proyectos y la ausencia de un marco formal de trabajo, proponiendo acciones para mitigarlos, como la capacitación en gestión de proyectos y la documentación sistemática de lecciones aprendidas.

En conclusión, la ejecución del plan de gestión del proyecto "*Downtime System*" permitió optimizar la gestión de eventos de líneas detenidas, incrementar la eficiencia operativa y fortalecer la toma de decisiones mediante el módulo de reportes. Además, se sentaron las bases para la estandarización y la mejora continua en el departamento Tec Ops. Como resultado del análisis y la experiencia adquirida, se recomendó mantener la metodología ágil, potenciar la capacitación en gestión de proyectos y ampliar la integración del sistema a otras áreas de la organización, promoviendo así una cultura de innovación y excelencia operativa.

ABSTRACT

The “Downtime System” project was carried out at the Heredia plant of Boston Scientific Costa Rica in response to the need to integrate and optimize the reporting systems for line downtime events, aligning with the Digital Transformation 2025 strategy. The main objective was to propose and develop a project management plan that, by applying best practices, would increase the likelihood of success in the system’s implementation, centralizing information and improving operational decision-making.

Project management was based on an agile approach using the Scrum framework but adapted according to the needs of the project and the organization. This allowed for incremental deliveries, continuous collaboration, and adaptation to frequent changes in requirements. Techniques such as focus group workshops, semi-structured interviews, and literature review were employed to gather requirements, identify best practices, and define the most suitable lifecycle. The management plan covered key areas of integration, scope, schedule, cost, quality, risk, communications, and change management, supported by collaborative tools like Exel, Jira and Miro.

Among the main results, the project was characterized, a prioritized backlog was defined, and tracking metrics were established, such as user stories, team velocity, the burndown chart, and the quality control dashboard. The main challenges identified for the graduation project were related to the lack of experience in project management and the absence of a formal work framework. To mitigate these, actions such as project management training and systematic documentation of lessons learned were proposed.

In conclusion, the execution of the “Downtime System” project management plan made it possible to optimize the management of line downtime events, increase operational efficiency, and strengthen decision-making through the reporting module. Furthermore, the foundations were laid for standardization and continuous improvement within the Tec Ops department. As a result of the analysis and experience gained, it was recommended to maintain the agile methodology, enhance project management training, and expand the integration of the system to other areas of the organization, thus promoting a culture of innovation and operational excellence.

INTRODUCCIÓN

El presente trabajo expone el desarrollo y gestión del proyecto "*Downtime System*" en el site de Heredia de Boston Scientific Costa Rica, como parte del pilar de Transformación Digital 2025. El objetivo principal es proponer un plan de gestión ágil que permita integrar y optimizar los sistemas existentes para el reporte y análisis de eventos de líneas detenidas en la planta, centralizando la información y mejorando la toma de decisiones operativas. A lo largo del documento se abordan el contexto organizacional, la problemática actual, los objetivos, el marco teórico y metodológico, así como el análisis de resultados y la propuesta de solución. Cada capítulo contribuye a la comprensión integral del proyecto, desde la caracterización de los requisitos hasta la definición de métricas y herramientas para el seguimiento y control, con el fin de asegurar la entrega exitosa del sistema y sentar precedentes para futuros proyectos en el departamento Tec Ops.

El documento está estructurado en capítulos que abordan de manera integral el desarrollo y gestión del proyecto "*Downtime System*":

- Capítulo 1: Generalidades de la investigación: Presenta el contexto organizacional, la problemática que motiva el proyecto, los objetivos generales y específicos, así como el alcance y las limitaciones del estudio.
- Capítulo 2: Marco teórico: Desarrolla los conceptos fundamentales sobre gestión de proyectos, metodologías ágiles, ciclos de vida de proyectos de software y herramientas tecnológicas relevantes para el contexto del proyecto.
- Capítulo 3: Marco metodológico: Explica la metodología utilizada para la investigación, incluyendo las técnicas de recolección y análisis de datos, la definición de categorías y variables, y la descripción de los productos esperados.
- Capítulo 4: Análisis de resultados: Expone los hallazgos obtenidos a partir de la aplicación de las herramientas metodológicas, detallando el análisis de requisitos, riesgos, lecciones aprendidas y la caracterización del proyecto.
- Capítulo 5: Propuesta de solución: Presenta la solución recomendada para el desarrollo e implementación del sistema, detallando el ciclo de vida seleccionado, la dinámica de trabajo del equipo, el plan de gestión y las herramientas a utilizar.

- Capítulo 6: Conclusiones y recomendaciones: Resume los principales aprendizajes y logros del proyecto, y sugiere acciones futuras para fortalecer la gestión de proyectos similares en la organización.
- Capítulo 7: Referencias bibliográficas: Incluye todas las fuentes consultadas y citadas a lo largo del trabajo, siguiendo las normas APA.
- Capítulos 8 y 9: Apéndices y Anexos: Contienen los instrumentos, plantillas, registros y materiales complementarios utilizados durante la investigación y el desarrollo del proyecto.

Capítulo 1 Generalidades de la investigación

El presente capítulo explica el contexto de la organización donde se realizará el trabajo. Se detalla la reseña histórica, su estructura organizacional, y marco estratégico, el tipo de proyectos que se ejecutan en el Departamento Tec Ops de *Boston Scientific* Costa Rica, así como el problema que este enfrenta, y que será atendido en este proyecto de investigación; mediante el planteamiento de objetivos, un alcance definido, e identificando aquellas limitaciones que podrían obstaculizar la mejora planteada por esta investigación.

1.1 Marco de referencia organizacional

En la presente sección se desarrolla la descripción de la empresa donde se ejecuta el proyecto.

1.1.1 Reseña histórica Boston Scientific.

La empresa *Boston Scientific* fue fundada el 29 de junio de 1979, en la ciudad de Boston, Estados Unidos. La compañía se estableció con la visión de desarrollar dispositivos médicos innovadores que mejorarían la salud del paciente con afecciones como las cardíacas, digestivas, pulmonares, vasculares, urológicas, de salud femenina y dolor crónico y su calidad de vida. Hoy en día cuenta con alrededor de 15 sedes de manufactura distribuidos en Costa Rica, Estados Unidos, Canadá, Israel, Irlanda, China y Malasia. Adicionalmente, la compañía ha sido reconocida por su primer trimestre de 2025. Esto refleja su éxito en la expansión global y su capacidad para ofrecer soluciones médicas innovadoras que transforman la vida de pacientes en más de 120 países, capacidad para adaptarse a las demandas del mercado y las presiones competitivas, con ventas netas operativas en mercados emergentes que crecieron 9.8% año tras año en el 2025

Desde su fundación a la actualidad su catálogo de productos se ha diversificado y a la fecha cuentan con dispositivos en las áreas de endoscopia, urología, intervenciones periféricas, cardiología intervencionista y electrofisiología. Esta gama de dispositivos permite la atención de más de 44 millones de pacientes cada año y emplea a más de 53 mil personas en sus diferentes plantas.

En Costa Rica la empresa *Boston Scientific* se estableció hace 20 años, abriendo su primer *site* en Heredia y debido a los buenos resultados se expandió, adquiriendo terrenos en el Coyol, Alajuela 2009 y en el 2024 abrieron un nuevo *site* en Cartago. Cabe destacar que la sede de Coyol se encuentra en las primeras tres en cuanto a tamaño y volumen. Mismo que cuenta con un nuevo

edificio de esterilización para dispositivos médicos y es el más grande bajo este concepto en *Boston Scientific*. La sede de Cartago está siendo preparado para abrir un edificio similar en tamaño, al edificio de Coyol.

Como se muestra en la **Figura 1.1**. Distribución de las sedes global, estas instalaciones forman parte de la red mundial de Boston Scientific, evidenciando la importancia estratégica de las sedes en Costa Rica dentro del mapa global.

Figura 1.1. Distribución de las sedes global



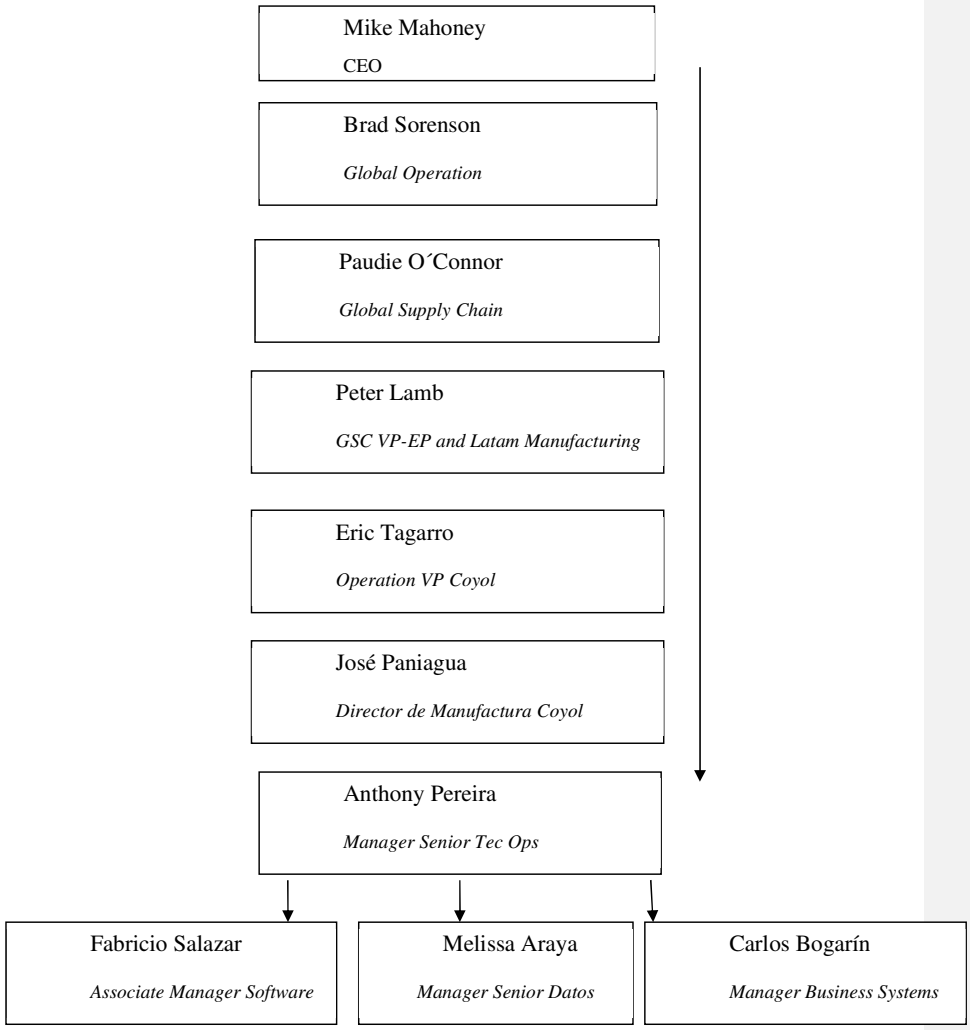
1.1.2 Estructura y marco estratégico.

A continuación, se presenta la estructura organizacional de la empresa Boston Scientific Costa Rica, donde se ubica el departamento *Technology for Operations* (Tec Ops).

1.1.2.1 Estructura organizacional.

En el organigrama representado en la **Figura 1.2**. Organigrama empresarial **s**e muestra la línea de reporte del departamento Tec Ops Costa Rica hacia el CEO Mike Mahoney

Figura 1.2. Organigrama empresarial



En la **Figura 1.2**. Organigrama empresarial se puede apreciar la ubicación del departamento de Tec Ops en el organigrama, así como sus tres aristas: ingeniería de datos, *software* y *Business Systems.*

El Departamento Tec Ops inició en el 2018, cuando el sistema *Manufacturing Execution* (MES) fue implementado en Coyol y Heredia, sustituyendo el uso de hojas de papel por un sistema para la gestión de la producción. El personal fue distribuido de la siguiente forma: 6 personas del equipo Tec Ops, las cuales se dedicaron a MES y 4 personas empleadas en la atención de requerimientos para creación de reportes y de *software.* Hoy en día, el equipo regional está conformado por 44 personas y dividido en tres aristas, como se ilustra en la **Tabla 1.1**. Aristas Tec Ops:

Tabla 1.1. Aristas Tec Ops

Business System	*Data Engineering & Advanced Analytics*	*Software*
23 personas	9 personas	12 personas

En el siguiente apartado se presenta el marco estratégico de la organización.

1.1.2.2 Marco estratégico.

La siguiente sección está compuesta por la misión, visión, valores y objetivos estratégicos de la organización.

- ***Misión.***

En Boston Scientific, nuestra misión es transformar vidas a través de soluciones médicas innovadoras que mejoran la salud del paciente.

- ***Visión.***

En Boston Scientific, nuestra visión es ser el líder en la transformación de vidas a través de soluciones médicas innovadoras.

- ***Valores.***

A continuación, se exponen los valores de la organización:

- Integridad.
- Innovación.
- Calidad.
- Colaboración.
- Cuidado del paciente.
- Diversidad e inclusión.

.

- ***Objetivos estratégicos.***

Los objetivos estratégicos de Boston Scientific son:

- Invertir en investigación y desarrollo para crear dispositivos médicos nuevos y mejorados.
- Colaborar con profesionales de la salud para comprender necesidades y desarrollar soluciones que satisfagan esas necesidades.
- Ampliar su presencia global para llevar sus productos y servicios a más pacientes en todo el mundo.

1.1.3 Proyectos en la organización.

El departamento Tec Ops está enfocado en atender las necesidades relacionadas con soluciones tecnológicas innovadoras, funcionales y adaptadas, además del análisis de datos que pueda surgir en los distintos departamentos de las tres sedes: Cartago, Coyol y Heredia; contribuyendo directamente a la transformación digital. Desde el año 2024 Tec Ops pasó a formar parte del Departamento de Ingeniería en la planta de Coyol.

Los proyectos atendidos por Tec Ops tienen los siguientes enfoques:

1- *Data Engineering & Advanced Analytics*: Su naturaleza es la ciencia y análisis de datos, mantenimiento y actualización de *datawarehouse*, mantenimiento y creación de reportes de las tres plantas, estudio de tendencias, gracias a *Neo4j, Power BI, Tableau,* las cuales son herramientas que se pueden integrar entre sí, para optimizar el análisis y la visualización de datos.

2- *Software*: Tiene como objetivo la creación de *software web* o de escritorio, a la medida.

3- *Business System*: Son de naturaleza de sistemas enfocado a la programación que necesita el sistema para visualizar la información de las áreas de producción.

A continuación, en el **Cuadro 1.1** Proyectos Tec Ops se recopilan los proyectos que se han ejecutado para diferentes departamentos, tales como: Producción, Calidad, Ingeniería, Equipos, Recursos Humanos, Finanzas, entre otros. Se debe tomar en cuenta que el **Cuadro 1.1** no es un portafolio, sino una recopilación ordenada de la información específicamente para la presentación en este trabajo.

Cuadro 1.1 Proyectos Tec Ops

Nombre del Proyecto	Descripción del Proyecto	Duración	Arista del departamento Tec Ops
FaraDigitalization	Colocar y realizar reportería de toda la información del producto *Farawave.* Se van creando los reportes conforme se va solicitando; pero es de prioridad alta todo lo que venga de *Farawave*, y que, al día de hoy, es el producto estrella de Boston Scientific.	1 año	*Data Engineering & Advanced Analytics*
LATAM Control Tower	Reportes financieros para medir *Spending, Labor, Scrap, PPV* y otros. Involucra *sites* de Costa Rica y el *site* de Missisagua, como parte de la estrategia de estandarización.	Desde el 2024- en proceso aún.	
Métricas de EHS	Reporte con métricas para controlar *Safety*, Emisiones de carbono.	4 meses	
Reporte *Headcount*	Reporte creado para controlar el *Headcount* en las plantas.	2 meses	
Reporte para eficiencia, *overtime*, *Women in Leadership*	Reportes creados para eficiencia, *overtime, Women in Leadership.*	2 meses	
Non Standard Production System	Sistema que controla y maneja la ejecución de ordenes no humanas en el piso de producción. Se sustituyó el uso y almacenaje de papel, por un sistema que controla y guarda los datos.	1 año	*Software Engineering*
Headcount system	Creación de un *software* para el control de *Headcount* Coyol, Heredia y Cartago.	8 meses	
SAC v2	*Software* utilizado por el personal directo de la planta, para el manejo de boletas de pago y todo lo referente a temas financieros.	5 meses	

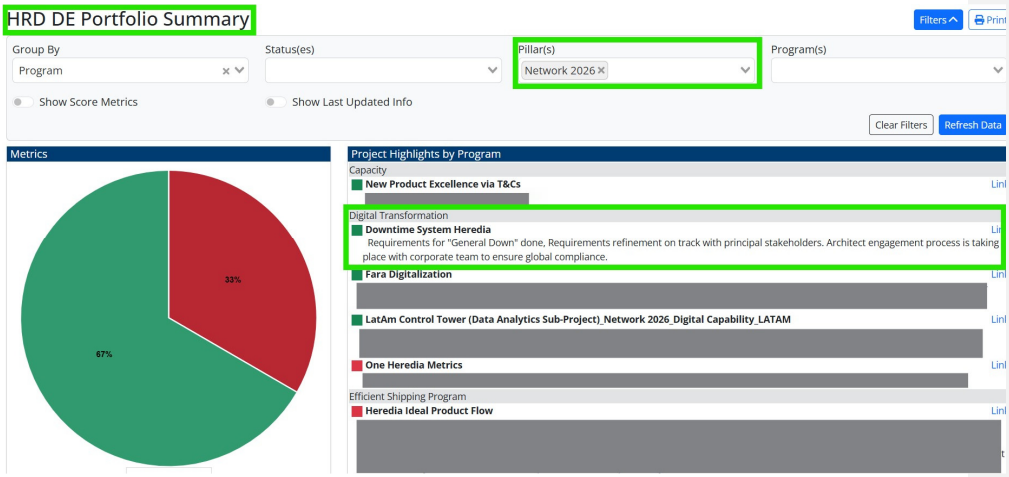
Nombre del Proyecto	Descripción del Proyecto	Duración	Arista del departamento Tec Ops
Lista *Master* de Químicos *Environmental Health & Safety*	Creación de un sistema para controlar el manejo de los químicos y fechas de vencimiento. Actualmente, solo se usa en Coyol.	2 meses	
Sealers connection to IPM	Conexión de las selladoras del piso de producción con la herramienta de IPM (*In-Process Monitoring)*	10 meses	
Impedance System	S*oftware* conectado con la máquina de *Impedance* que envía el resultado de la prueba a MES, evitando datos manuales.	1 año y 1 mes.	
Deprag Automation Software	Conexión del controlador *Deprag Exalt Model B*, del piso de producción con MES para evitar el ingreso manual de datos.	5 meses	
Automated Serial Number Reading System	*Software* capaz de leer el serializado desde el sistema *Eprom* y pasarlo al sistema de MES, evitando datos manuales.	2 meses	
Tomes	Modelado y mantenimiento del área, de acuerdo con las solicitudes de ingenieros de Manufactura, calidad y producción.	4 meses	*Business Systems*
Snares		5 meses	
GBF		5 meses	

Nombre del Proyecto	Descripción del Proyecto	Duración	Arista del departamento Tec Ops
Sensor		3 meses	
Exalt		3 meses	
Interject		3 meses	
Farawave		5 meses	
Stents		6 meses	

1.1.4 Proyecto Downtime System

En el año 2024, durante el *Development Excellent* (DE) del *site* de Heredia, en el mes de diciembre (reunión mensual de presentación de resultados). En esta reunión presentaron también los proyectos estratégicos que se implementarán en el año 2025 tales como: los proyectos *Star* (proyectos a ejecutar *on site* 2025) para el Pilar de *Network 2026* dentro del programa *Digital Transformation*, además se formalizó el entregable del Proyecto *Downtime* y su seguimiento formal dentro del portafolio del DE, como se muestra en la **Figura 1.3**. Panel DE Heredia.

Figura 1.3. Panel DE Heredia



Nota: Las áreas en gris se consideran confidenciales para la empresa.

Este proyecto busca integrar tres sistemas que actualmente se encuentran desconectados (dos hojas de cálculo Excel y un sistema interno) y que son utilizados por distintas áreas para gestionar eventos de *downtime*, eliminando redundancias. Actualmente se utilizan los sistemas:

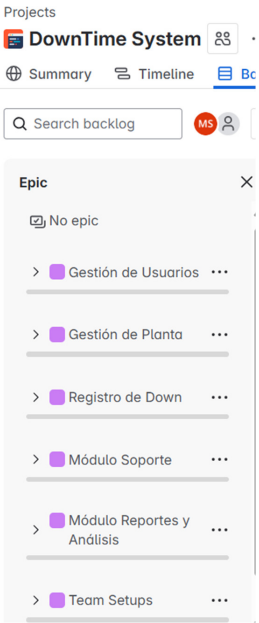
- VTP (departamento *Supplier Quality*): Utilizado para registrar tiempos de parada y cantidad de unidades afectadas manualmente. Temas referentes a materia primas, bodega y otros.
- *Production Tracking System (PTS)* (departamento Equipos): Sistema de incidentes para reportar fallas o paradas en los equipos utilizados en las líneas de producción.

- *Line Disruption* (departamento Ingeniería Industrial): Hoja de cálculo usada para registrar tiempos de parada por operación y estimar unidades impactadas.

El proyecto consiste en la integración de estos sistemas, consolidando los datos en una sola base de datos y manteniendo su conexión con otros sistemas de la compañía (extraer información de equipos, extraer información de MES, extraer información de los usuarios). Además, debe mejorar lo que hacen los sistemas actuales por solicitud del cliente, creación de método para envíos de notificaciones de correo, cálculos automáticos que generarán la data que sustituye PTS, manejo de roles y todos los mantenimientos necesarios para que los administradores tengan autonomía en su nuevo sistema.

El proyecto tiene sus fases a alto nivel definidas según la **Figura 1.6**. Plan *Downtime,* pero dentro de la herramienta Jira, el proyecto está siendo dividido para el desarrollador de acuerdo con la **Figura 1.4**. Procesos en Jira *Downtime.*

Figura 1.4. Procesos en Jira *Downtime*



En la actualidad el proyecto ha cubierto las siguientes actividades:

- Reunión de Inicial (conocer problema, interesados, análisis de otros sistemas que usa Boston Scientific).
- Reunión de propuesta (diagrama de la solución propuesta)
- Recolección de requisitos mínimos del sistema.
- Tareas de validación *Global System Methodology* (GSM).
- Primea reunión con el *Arquitecture Engagament* (equipo corporativo encargado de alineamiento de tecnologías).

No obstante, el proyecto presenta como pendientes, las siguientes tareas:

- Plan de riesgos.
- Cronograma de sus tareas.
- Refinamiento de los requisitos.
- Programación 0% avanzado.
- Calidad de *software* 0% avanzado.

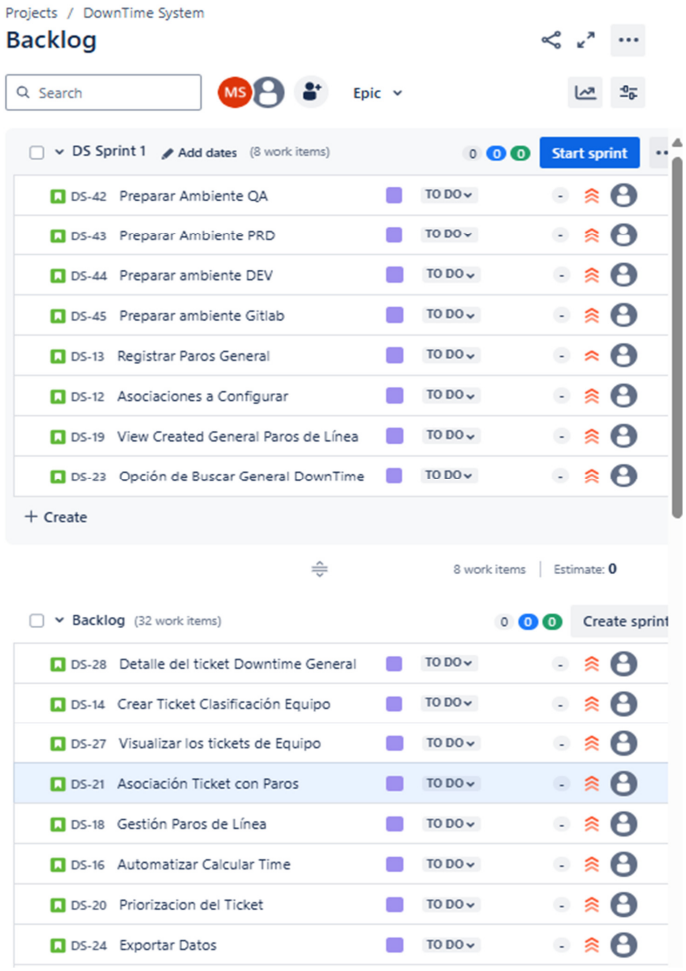
Los usuarios finales de este sistema son: el grupo de operaciones de la planta, departamento de materiales, e industrial y equipos; mientras que, sus administradores serán personas específicas de esas áreas.

Algunos beneficios esperados de la solución son los siguientes:

- Eliminación de redundancia: Un único punto de entrada alimentará todos los módulos, reduciendo errores y duplicidad.
- Trazabilidad de eventos: Capacidad de seguir un evento desde su origen hasta su resolución, mediante referencias cruzadas.
- Base de datos unificada: Facilita mantenimiento, integridad de datos y soporte.
- Cálculo automático del impacto: Reducción de tiempo y errores en el análisis de los paros de línea.
- Visualización gráfica: Representación digital del estado de líneas y equipos para facilitar la toma de decisiones.
- Análisis de datos y tendencias: Mejora continúa basada en datos reales.
- Soporte local: Gestión y mantenimiento continuo, por parte del equipo Tec Ops.

Algunos requisitos del sistema a desarrollar se presentan en la **Figura 1.5**. Requerimientos en definición *Downtime system*:

Figura 1.5. Requerimientos en definición *Downtime system*



1.2 El problema y su impacto

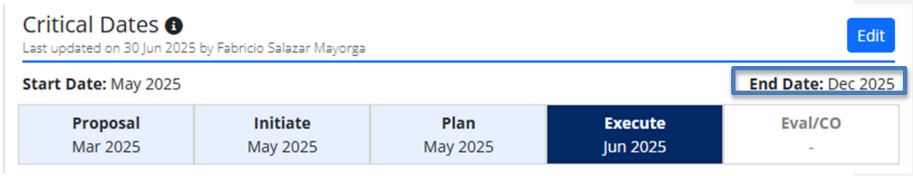
La sede de Coyol de Boston Scientific Costa Rica, específicamente el departamento de *Technology for Operations* (Tec Ops) arista de *Software*, tiene a su cargo el desarrollo del proyecto

Downtime; el cual actualmente se encuentra retrasado con respecto a las metas esperadas; representando un problema para el departamento Tec Ops, el cual debe ser atendido a la brevedad. El retraso de este proyecto en particular se asocia con lo que ya ocurre en la organización, esto es, los proyectos finalizan con retraso. Esta situación revela que es probable que este proyecto también termine tarde.

Como se observa en **Figura 1.6**. Plan *Downtime* el proyecto debería estar en ejecución, y apenas se encuentra en la etapa de propósito, con lo cual, tiene un estimado de 2 meses de retraso; para un proyecto cuya duración es 5 meses, o sea, ya suma el 40% de atraso y aún no se ha terminado dicha etapa. Si no hubiera retraso, el proyecto terminaría por calendario en noviembre, pero dado el estado actual, estaría terminando en diciembre, lo que supone es su fecha máxima de entrega para el Pilar de Transformación Digital. Ver en **Figura 1.6**. Plan *Downtime*en el "*End Date*: *Dec* 2025".

Comentado [AHG1]: Agregar referencia cruzada.

Figura 1.6. Plan *Downtime*



Comentado [AHG2]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

La primera causa es que el equipo de Tec Ops cuenta con recursos técnicos con conocimientos básicos en gestión de proyectos. Estos colaboradores asumen el rol de PM, de forma empírica, utilizando sus propios criterios para definir los planes, documentación y enfoques de seguimiento. Para el proyecto *Downtime* se conoce que el desarrollador del sistema está tomando un rol dual, siendo *Product Owner* (PO) del proyecto, sin disponer del conocimiento necesario en gestión de proyectos, de acuerdo con lo conversado con él (A. Gutierrez, comunicación personal, 4 de julio de 2025). En la **Tabla 1.2.** Personal con conocimiento en gestión de proyectos., se puede observar que, al departamento, en su arista de *software,* aún le falta conocimiento en gestión de proyectos, ya que, solo una persona (es decir 8,33%) está certificada en Scrum y otras 3 tienen experiencia en gestión, lo que equivale al 25%.

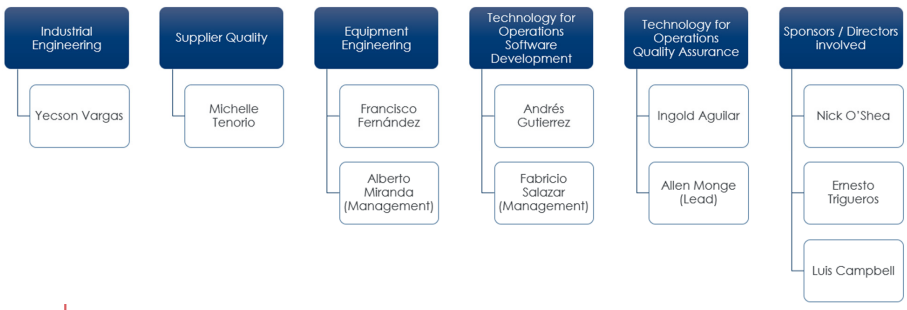
Tabla 1.2. Personal con conocimiento en gestión de proyectos

Arista	Certificación Scrum	Otra Certificación de Gestión de proyectos o experiencia previa
Business Systems	4 de 23 = 17,39%	0
Data Engineer	3 de 9 = 33,33%	3 de 9 = 33,33%
Software	1 de 12 = 8,33%	3 de 12= 25%

Comentado [AHG3]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

Una segunda causa que hace que el proyecto se encuentre retrasado, así como los entregables esperados, es que el mismo no tiene un marco formal de trabajo que indique lo que le corresponde a cada rol. Esto hace que los interesados no tengan claridad sobre sus roles, responsabilidades y el nivel de compromiso esperado durante el ciclo de vida del proyecto. El equipo de *software* realizó para el *Downtime System* una reunión de "Inicio", donde se mencionaron los interesados, sin establecer sus funciones, ver **Figura 1.7.** Interesados *Downtime***.** La participación de los interesados en las reuniones depende de que no tengan contratiempos en sus áreas, por lo que se percibe que el proyecto no avanza en el recopilamiento de requerimientos, de acuerdo con lo esperado. Esto se explicó en la **Figura 1.6**. Plan *Downtime*, donde se muestra que la ejecución del proyecto debió iniciar en junio, pero a julio 2025, este no ha iniciado con su desarrollo.

Figura 1.7. Interesados *Downtime*

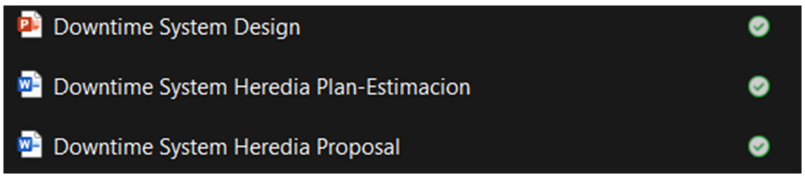


Comentado [AHG4]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

Y la última causa que se identificó, se relaciona con la forma como el departamento de *Technology for Operations* (Tec Ops) realiza el control de cambios. Esto se cataloga como

empírico porque va asociado a que el departamento no tiene un marco formal en gestión de proyectos. Por lo tanto, no se tiene un elemento formal que indique cómo deben gestionarse los cambios y elementos críticos. Según la investigación realizada a la documentación del proyecto *Downtime* (**Figura 1.8**. Documentos *Downtime*), no existe documentación de un control de cambios, por lo tanto, el proyecto está siguiendo el mismo patrón que ha causado demoras en la entrega de proyectos anteriores del departamento.

Figura 1.8. Documentos *Downtime*



Comentado [AHG5]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

Si se siguen dando retrasos del proyecto, el primer impacto sería la afectación en el desempeño de la sede de Heredia. Dado que el proyecto es parte del pilar de Transformación Digital y se ha comprometido la entrega de su producto mínimo viable (MVP) para el 2025, todo retraso afectará el cumplimiento de la estrategia organizacional y las metas de eficiencia operativa del *site*. En la **Figura 1.3.** Panel DE Heredia se puede observar que el proyecto está dentro de la estrategia de la sede para el 2025 y, además, en la **Figura 1.6**. Plan *Downtime* se puede observar la fecha comprometida 2025.

Un segundo impacto derivado de la continuación del retraso en la entrega del proyecto es el efecto negativo directo sobre las operaciones del área de producción. Actualmente, los colaboradores presentan incertidumbre respecto al proceso correcto de ingreso de información a los Excel (VTP y *Line Disruption Tool*) o sistema (*Production Tracking System*), lo que genera confusión y errores frecuentes, en cuanto a la categorización de incidentes. Esto ha provocado que, en múltiples ocasiones, se registren solicitudes en áreas incorrectas o no se registren del todo, como se muestra en la **Tabla 1.3**. Incidentes mes de junio 2025, donde por proceso, ambos sistemas deberían tener la misma cantidad de incidentes reportados, pero revisando el mes anterior, el 10% de información en *Line Disruption* (herramienta de industrial) no fue reportada de acuerdo con los reportado en *Production Tracking System* (herramienta de equipos).

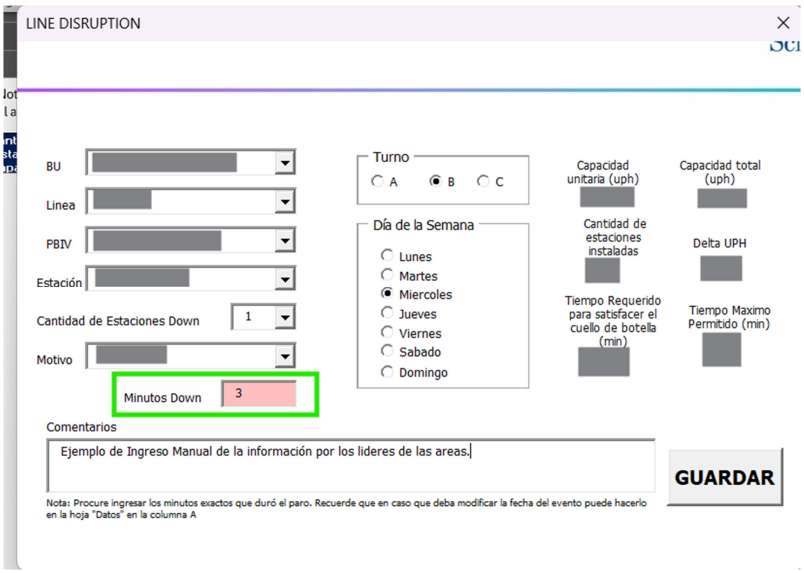
Tabla 1.3. Incidentes mes de junio 2025

Incidentes *Production Tracking System*	*Line Disruption* Reportados
200	120

Comentado [AHG6]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

Por último, si el sistema *Downtime* continúa con estos retrasos el personal de los departamentos de industrial, materiales y equipos deberán continuar utilizando sus herramientas que presentan varias deficiencias: no existen las notificaciones de correo; en materiales problemas de seguimiento a vendedores de materia prima, no tienen roles definidos de forma automática, el tiempo de un paro es ingresado manualmente, como se observa en la **Figura 1.9**. Ingreso Manual *Line Disruption* por tanto no es información confiable para toma de decisiones, los incidentes reportados de manera incorrecta afectan los reportes. Por lo tanto, continuar utilizando sus herramientas conlleva más trabajo para estos departamentos y no tienen claridad en sus reportes realmente.

Figura 1.9. Ingreso Manual *Line Disruption*



Nota: Las áreas en gris se consideran confidenciales para la empresa.

1.3 Objetivos

A continuación, se presenta el objetivo general y los objetivos específicos del proyecto.

1.3.1 Objetivo general.

Proponer un plan de gestión para el proyecto "*Downtime System*", a partir de la caracterización de sus componentes clave y la aplicación de buenas prácticas en gestión de proyectos, para el incremento de las probabilidades de éxito en su ejecución.

1.3.2 Objetivos específicos.

1. Caracterizar el proyecto "*Downtime System*", mediante el levantamiento de requerimientos funcionales y no funcionales, para la determinación de su alcance y funcionalidades esperadas.

2. Identificar buenas prácticas en gestión de proyectos, mediante un estudio basado en los requisitos del *Downtime system*, para la confirmación de efectividad en el contexto del proyecto.

3. Desarrollar un plan de gestión del proyecto que incluya los planes subsidiarios necesarios, para la maximización de las probabilidades de éxito y entrega oportuna del sistema.

4. Establecer mecanismos de seguimiento y evaluación, mediante el uso de métricas claves que permitan la toma decisiones correctivas a tiempo durante la ejecución del proyecto.

1.4 Alcance y limitaciones

A continuación, se presentan el alcance y las limitaciones del presente proyecto.

1.4.1 Alcance

El alcance del proyecto será la entrega de un plan de gestión ágil para el proyecto *Downtime System*, que sea adecuado y funcional para el departamento de Tec Ops. Otros proyectos del departamento quedan fuera del alcance del presente proyecto. Entonces, se continua a enlistar la lista de sub-entregables que generará el proyecto:

El primer entregable del proyecto es el alcance del proyecto, el cual está compuesto por: el Acta de constitución del proyecto, documento con los requerimientos, lista de tareas inicial priorizado y estimada. Para la elaboración de estos productos se tendrán como insumos, la entrevista y observación directa, para conocer requerimientos y expectativas de los clientes y determinar así, las funcionalidades necesarias y delimitar el alcance.

El segundo entregable es la lista de buenas prácticas que se aplicaron durante la gestión del *Downtime System*. Se plantea la entrega de una lista de todas las buenas prácticas estudiadas y en prosa se explicará la escogencia de estas para el proyecto y la lista final que fue aplicada.

El tercer entregable es el plan de gestión ágil para el proyecto, donde se incluyen los planes subsidiarios, herramientas, artefactos y técnicas seleccionadas y la estrategia de control de los planes: plan de gestión de integración, alcance, tiempo, costo, calidad, interesados y comunicaciones, riesgos y cambios.

Por último, la entrega de un tablero de métricas, que permita dar seguimiento y evaluar el estado del proyecto para tomar decisiones claves. Para la elaboración de las métricas se plantea ejecutarlas y monitorearlas desde el arranque del proyecto, tal es el caso de: métricas de cronograma, métricas de alcance/ entregable, métricas de recursos/ equipo, métricas de calidad, métricas de riesgos.

Los entregables del proyecto conforman el alcance del proyecto y además pretenden aportar al departamento de Tec Ops, en términos de gestión para la implementación y éxito del sistema requerido. Además, se pretende sea un precedente para que los futuros proyectos sean ejecutados bajo el mismo estándar y mejorar en tiempos de entrega.

1.4.2 Limitaciones.

Falta de disponibilidad de los interesados para evacuar dudas o aplicar herramientas.

Documentación escueta de proyectos anteriores para los análisis adecuados.

Capítulo 2 Marco teórico

En el presente capítulo se desarrollan los conceptos básicos, los complementarios y los específicos necesarios para el entendimiento de la temática que se investiga. Para ello, se parte del concepto de proyectos en general para luego enfocarse en software así se facilita la compresión de desde lo general a lo especifico. Se profundiza en la gestión, ciclos de vida, marcos de trabajo, practicas actuales en software y herramientas tecnológicas.

2.1 Proyectos de Software y su gestión

En la siguiente sección se abordarán diversos conceptos básicos y fundamentos en gestión de proyectos que se permitirán alineamiento con el tema de estudio.

2.1.1 Definición de proyecto.

Según la ISO 10006:2017, proyecto se define como un proceso único que se emprende para lograr un objetivo. Se amplia definiendo proceso como conjunto de actividades coordinadas y controladas, con fechas de inicio y Finalización. (ISO 10006, 2017).

Para el PMI® y según el PMBoK® séptima edición, es un artefacto producido, cuantificable y que puede ser en sí mismo un elemento terminado o un componente de un elemento. (PMBoK® , 2021).

Considerando los conceptos establecidos por la ISO y el PMI, es posible identificar con claridad qué constituye un proyecto y qué no. Este conocimiento resulta especialmente relevante, ya que muchas organizaciones desde pequeñas empresas hasta grandes corporaciones ejecutan iniciativas que cumplen con las características de un proyecto, aunque no siempre las reconozcan formalmente como tales.

La ejecución de proyectos es una necesidad estratégica para las compañías, ya que permite implementar mejoras o corregir procesos con el objetivo de generar beneficios, ya sea mediante el aumento de ingresos, la reducción de gastos o el fortalecimiento de la eficiencia y la eficacia operativa.

En este contexto, el caso en estudio busca precisamente mejorar la eficacia y eficiencia de ciertos departamentos, además de estandarizar el proceso en las tres plantas ubicadas en Costa Rica.

Sin embargo, cuando las organizaciones no cuentan con un nivel de madurez adecuado en gestión de proyectos, es común que se confunda el concepto de proyecto con las tareas propias de

la operación diaria. Por esta razón, resulta fundamental comprender las características que definen a un proyecto.

De acuerdo con OBS *Business School* (escuela de negocios *born online* en español) (Pérez, 2021) en su blog "Tipos de proyectos y sus principales características", se enumeran las siguientes características:

- Son únicos: aunque apliquemos pasos estándares para llevarlos a cabo, siempre será necesario ajustarlos y personalizarlos. Salir de la rutina y enfrentarse a lo desconocido.
- Cuentan con un tiempo limitado: Es necesario establecer fechas de entrega de cada objetivo y a la medida de lo posible ir cumpliendo el plazo.
- Necesita un equipo: Además, este equipo suele ser diferente por proyecto. Se establecen roles y se reparten tareas dependiendo de las características de la persona.
- Cuenta con unas fases que son flexibles: Para sacar adelante un proyecto, como mínimo es necesario planificarlo, ejecutarlo y entregarlo. Estas fases suelen subdividirse en sub-fases que son flexibles.
- Tienen unos elementos clave: Estos elementos clave componen su estructura y además permiten que puedan llevarse a cabo.

2.1.2 Proyectos de software.

La necesidad de sistemas ha ido en aumento en un mundo donde la tecnología se ha convertido en la esencia del progreso humano. Ante cualquier conflicto operativo o de gestión, surge la propuesta de implementar un software como solución: evitar el error humano y facilitar la ejecución de tareas de manera más simple y eficiente. Sin embargo, la ejecución de proyectos de *software* ha sido históricamente cuestionada, principalmente por los cambios que surgen durante su desarrollo. Esto ha generado un debate constante sobre cuál metodología de gestión es más adecuada: ¿cascada o ágil? Este tema será abordado en detalle más adelante.

El enfoque de estudio se centra en la construcción de software. Según la Revista Cubana de Ciencias Informáticas (Delgado y Días, 2021), solo una pequeña parte de los proyectos de software desarrollados finaliza dentro de los plazos y costos establecidos, cumpliendo además con

los requerimientos acordados. Esto evidencia que los sistemas representan un desafío complejo, especialmente porque los clientes o usuarios finales suelen tener expectativas muy altas o tienden a omitir detalles que pueden modificar los requerimientos iniciales.

2.1.3 Gestión de los proyectos.

La gestión de proyectos se refiere a la dirección y coordinación de iniciativas o esfuerzos con el fin de alcanzar objetivos o metas específicas, que a menudo implican lidiar con riesgos. Sin embargo, a través de los años la forma de abordar esta gestión a cambiado con los años, gracias al interés de evolucionar y poner en práctica ajustes al concepto para adoptar un enfoque que sea ideal independientemente el proyecto a gestionar.

La gestión de proyectos es un componente organizacional de decisiones importantes para cualquier departamento. Y según la *Revista de Ciencias Sociales* (Meléndez y El Salous, 2021) tiene factores críticos como son: costo, tiempo, calidad y capacidad gerencial. Se busca optimizar sus operaciones reduciendo costos, mejorando tiempos y calidad de producto, pero también debe existir ese respaldo en la ejecución de los proyectos por medio de la gerencia.

¿Hacía donde lleva una adecuada gestión de proyectos a una organización?

Se dirige al éxito organizacional acompañado de mejora continua de procesos incluyendo calidad, eficiencia y eficacia entre sus pilares. Se pueden identificar los beneficios para una organización; la estructuración de actividades, el aprovechamiento y asignación de los recursos y plazos definidos para la entrega de procesos y alineamiento con su estrategia. Todo esto no sería posible sin la tecnología y las herramientas actuales que permiten control y visualización del equipo del proyecto (más adelante se profundizará en el tema).

Luego de comprender los beneficios de una adecuada gestión de proyectos, es importante considerar también los desafíos comunes que enfrentan las organizaciones. Entre los problemas más frecuentes se encuentran:

- Plazos de entrega con muchos cambios.
- Rehacer el trabajo.
- Interrupciones en el ritmo de trabajo.
- Cambios en el alcance.
- Planificación insuficiente.

- Control inadecuado.
- Aumento de los costos.
- Problemas de comunicación.

Esta información ha sido extraída de Gerencia de Proyectos (Lanús, 2016), y refleja una realidad que ha llevado a cuestionar los enfoques tradicionales de gestión. En este contexto, surgen las metodologías ágiles como una alternativa que ofrece mayor flexibilidad ante los cambios. No obstante, estas metodologías también enfrentan sus propios retos, los cuales serán abordados más adelante.

La dirección y gestión de un proyecto representa una labor compleja que exige del *Project Manager* (PM) un conjunto sólido de habilidades, conocimientos técnicos y una alta capacidad de organización. En el desarrollo de cualquier proyecto, se deben coordinar múltiples tareas que requieren planificación detallada, asignación de recursos, control de tiempos, gestión de riesgos y comunicación efectiva entre los involucrados.

Es fundamental reconocer que cada proyecto tiene características únicas, por lo que no existe una fórmula universal para garantizar el éxito. La clave está en aprender de los errores cometidos, aplicar ese conocimiento en futuras iniciativas y así avanzar progresivamente hacia una gestión más eficiente y cercana a la excelencia.

2.1.3.1 Ciclo de vida

Según el *PMBoK®* su definición corresponde a una seria de fases que atraviesa un proyecto desde su inicio hasta su conclusión (*PMBOK®*, 2021).

Según la norma ISO 21500, acá se describe las fases típicas que un proyecto debe atravesar, desde la concepción hasta la finalización de este. Esto incluye la definición de necesidades, planificación, ejecución, monitorización, control y cierre del proyecto (ISO 21500, 2021).

Ambas definiciones la del ciclo de vida de un proyecto según el *PMBOK* y la establecida por la norma ISO comparten similitudes y no se limitan exclusivamente al ámbito de gestión de proyectos. De hecho, pueden aplicarse a tareas cotidianas, como pintar una casa, donde existe una secuencia lógica de pasos que deben seguirse para lograr una ejecución ordenada y exitosa. Estas metodologías, aunque formalizadas en contextos profesionales, tienen una utilidad práctica en la vida diaria, al ofrecer estructura y claridad en la realización de cualquier actividad.

Sin embargo, en el entorno laboral, el concepto de ciclo de vida de un proyecto ha evolucionado significativamente. A medida que los proyectos se han vuelto más complejos y diversos, también lo han hecho los enfoques para gestionarlos. No todos los proyectos pueden abordarse de la misma manera, ya que cada uno presenta características, riesgos y objetivos particulares. Por ello, así como la tecnología ha avanzado, también lo han hecho los conceptos clave en la gestión de proyectos.

Esta transformación ha dado lugar a nuevas formas de entender y aplicar los ciclos de vida, adaptándolos a las necesidades específicas de cada iniciativa. En el siguiente apartado del documento se explorará cómo estos conceptos han evolucionado y cómo se han adaptado a los distintos tipos de proyectos, permitiendo una gestión más flexible, eficiente y alineada con los objetivos estratégicos de las organizaciones. En la **Cuadro 2.1**. Características en los ciclos de vida, se puede observar un resumen con características y los diferentes ciclos de vida es una imagen modo resumen de lo que se explicara en detalle en cada ciclo de vida.

Cuadro 2.1. Características en los ciclos de vida

Enfoque	**Requisitos**	**Actividades**	**Entrega**	**Meta**
Predictivo	Fijos	Realizados una única vez para todo el proyecto.	Entrega única.	Gestionar costos.
Iterativo	Dinámicos	Repetidos hasta que este correcto.	Entrega única.	Corrección de la solución.
Incremental	Dinámicos	Realizados una vez para un incremento dado.	Entregas frecuentes y más pequeñas.	Velocidad
Ágil	Dinámicos	Repetidos hasta que este correcto	Entregas pequeñas frecuentes.	Valor para el cliente mediante entregas frecuentes y retroalimentación.

2.1.3.2 Ciclo de vida predictivo

Este enfoque tradicional y se caracteriza por seguir una secuencia lineal y estructurada en la ejecución del proyecto. Se aplica principalmente cuando los requisitos pueden definirse, recopilarse y analizarse de forma clara desde el inicio, lo que permite establecer una planificación detallada y estable.

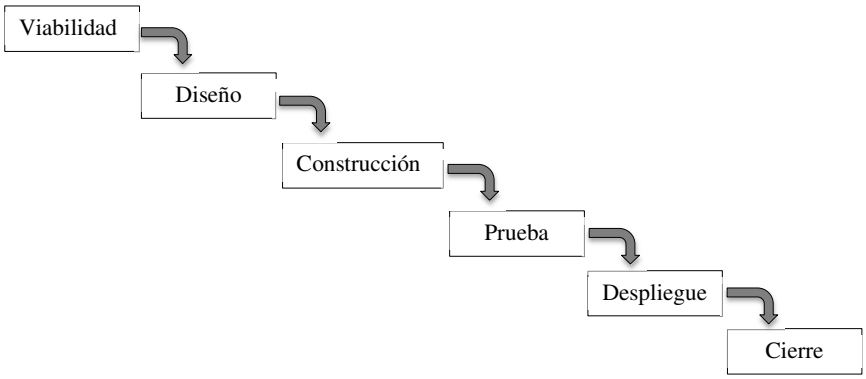
Según el *PMBOK® Guide* (2021), este ciclo es especialmente útil en proyectos donde la inversión es considerable y los riesgos son significativos, lo que exige un alto nivel de control, seguimiento y revisión constante. Además, contempla mecanismos formales para el control de cambios, permitiendo que ciertas fases del proyecto puedan ser replanteadas si es necesario, aunque esto suele implicar un proceso más riguroso y menos flexible que en metodologías ágiles.

Durante la aplicación en cascada, es común que elementos como el alcance, el cronograma, el presupuesto, las necesidades de recursos y los riesgos queden bien definidos desde el inicio. Esto reduce el nivel de incertidumbre y facilita la gestión, especialmente cuando se pueden aprovechar plantillas o modelos de proyectos anteriores con características similares.

Resulta ideal en contextos donde los requisitos son estables, el entorno es predecible y se requiere una documentación exhaustiva. Sin embargo, su rigidez puede representar una limitación en proyectos que demandan adaptabilidad o que están sujetos a cambios frecuentes, lo que ha impulsado la adopción de metodologías más flexibles como las ágiles, que serán abordadas en secciones posteriores.

Según el *PMBOK® Guide* (2021), este ciclo de vida consta de seis fases principales, ver **Figura 2.1**. Ejemplo de ciclo de vida predictivo, cada una con objetivos específicos que contribuyen al desarrollo ordenado del proyecto. Estas fases permiten una planificación rigurosa y un control detallado, lo cual es ideal en contextos donde los requisitos son estables y bien comprendidos desde el inicio, además solo se pueden continuar con la siguiente fase si la anterior está finalizada.

Figura 2.1. Ejemplo de ciclo de vida predictivo



Las seis fases del ciclo de vida predictivo son:

1. Viabilidad: Se determina si es válido el negocio y si la organización tiene capacidad para entregar el resultado previsto.
2. Diseño: Según su planificación y el análisis se llega a un prototipo del diseño final.
3. Construcción: Se construye el entregable según los diseños aprobados.
4. Prueba: Revisión de calidad final e inspección del entregable, así como la aceptación del cliente.
5. Despliegue: Se pone en uso el entregable final y actividades necesarias para transición y aceptación del producto final.
6. Cierre: El cierre del proyecto, conocimiento y los artefactos se guardan y recursos o equipo del proyecto son liberados y contrato cerrado.

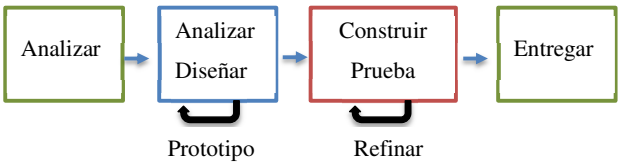
2.1.3.3 Ciclo iterativo

Este ciclo se caracteriza porque los requisitos del proyecto tienen un alto nivel de incertidumbre, volátiles y muy posible cambian a la largo del proyecto. La visión del proyecto y los requisitos iniciales conocidos son refinados, detallados y pueden cambiar o ser remplazados a

lo largo del proyecto por los comentarios del usuario (se muestran prototipos o pruebas de concepto al cliente).

Durante este ciclo, las iteraciones entregables cortos del producto del proyecto son breves, ya que la retroalimentación del cliente es fundamental para implementar cambios cuando sea necesario y así cumplir con sus exigencias. Este ciclo iterativo se caracteriza por una alta participación del cliente y una dinámica de cambios constantes. Esto se ilustra en la **Figura 2.2**. Ciclo de vida iterativo se evidencia cómo se van adaptando los cambios luego de la retroalimentación del cliente.

Figura 2.2. Ciclo de vida iterativo

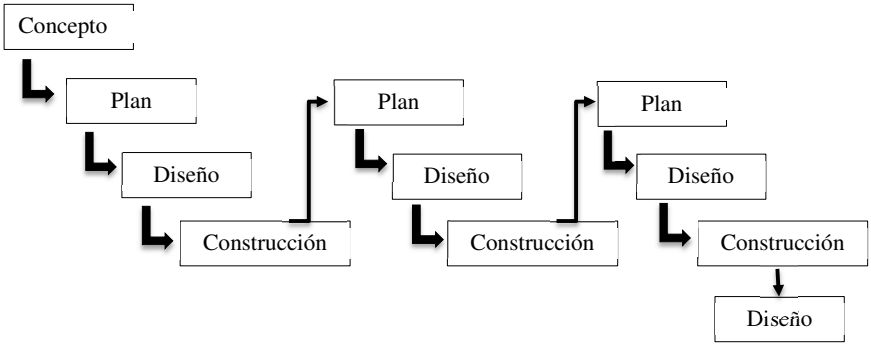


Nota: Adaptada de "Guía de Práctica Ágil" (p. 21), por Project Management Institute, 2018.

2.1.3.4 Ciclo de vida incremental

El concepto hace referencia a que la idea general de lo que quiere el cliente es decir su producto final esta conceptualizado, pero se van desarrollando características y funciones en iteraciones cortas para mostrar al cliente, recibir retroalimentación y cada entrega corta es puede ser utilizada por el cliente, **Figura 2.3.** Ciclo vida incremental, se desarrolla una casa y cada entrega corta ya puede ser utilizada por el cliente. Tiene la característica que sus fases se pueden repetir durante en distintas etapas del proyecto tantas veces sea necesario.

Figura 2.3. Ciclo vida incremental



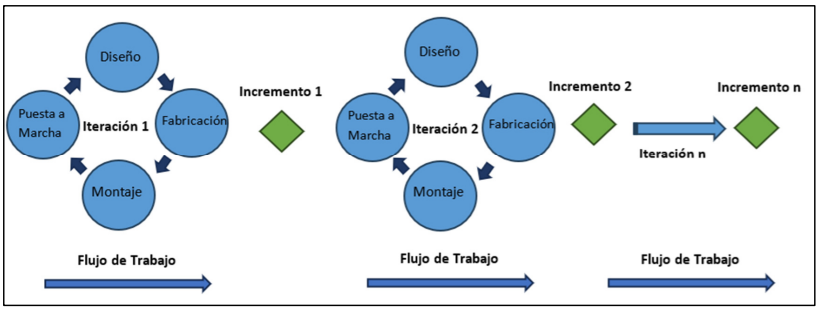
Nota: Adaptada de "Guía de Práctica Ágil" (p. 22), por Project Management Institute, 2018.

2.1.3.5 Ciclo de vida ágil

Este ciclo se basa en el ciclo iterativo y el incremental, reduciendo el desperdicio y el retrabajo. Cada una de estas iteraciones es conocida como iteración, al final de cada iteración, el usuario revisa el entregable y se brinda la retroalimentación, ver

Figura 2.4. Ciclo vida ágil para detalle. Esta retroalimentación es esencial, ya que permite realizar ajustes oportunos y fomentar la mejora continua del producto. Gracias a este enfoque, el equipo de desarrollo puede adaptarse rápidamente a cambios en los requisitos, prioridades o expectativas del cliente, asegurando que el resultado final se alinee con sus necesidades reales.

Figura 2.4. Ciclo vida ágil



Nota: Adaptada de "Guía de Práctica Ágil" (p. 24), por Project Management Institute, 2018.

2.2 Practicas actuales en desarrollo de Software

Según Revista multidisciplinar g-ner@ndo isnn: 2806-5905 (2024), existen tres secciones que deben ser analizadas cuando se pregunta por prácticas actuales en desarrollo de software, tendencias, metodologías y desafíos. Se ampliará en temas como adopción de prácticas agiles, automatización con *DevOps*, la incorporación de inteligencia artificial (IA) en el desarrollo y evolución hacia la arquitectura de la nube y sistemas distribuidos.

Dado el análisis previo sobre el ciclo de vida de los proyectos y la investigación realizada, se puede afirmar que actualmente las metodologías ágiles han ganado un protagonismo significativo en el ámbito del desarrollo de *software*. Este auge responde principalmente a la

naturaleza dinámica y cambiante de los proyectos tecnológicos, donde los requerimientos pueden evolucionar rápidamente conforme avanza la ejecución.

Las metodologías ágiles permiten una mayor flexibilidad. Estas promueven entregas incrementales, retroalimentación continua y adaptación constante, lo que resulta especialmente valioso en entornos donde la incertidumbre y el cambio son frecuentes. Pero el desafío más grande acá es aplicar la metodología, vivirla y no solo quedarse con su estructura textual.

En este contexto, la agilidad no solo mejora la capacidad de respuesta ante nuevas necesidades del cliente, sino que también fomenta una colaboración más estrecha entre los equipos de trabajo, incrementando la eficiencia y la calidad del producto final.

La adopción de prácticas *DevOps* se ha consolidado como un estándar en muchas organizaciones modernas, especialmente en aquellas orientadas al desarrollo ágil de software. *DevOps* permite una integración y despliegue continuo de aplicaciones de manera rápida, eficiente y confiable, lo que representa una ventaja competitiva en entornos altamente dinámicos. Mas adelante se abordará más sobre la herramienta de *DevOps*.

Muy relacionado con los principios del agilismo, se destaca el creciente uso de la inteligencia artificial (IA) como una herramienta clave para optimizar tareas dentro de la ingeniería de software. La IA ha demostrado ser especialmente útil en actividades como el análisis y optimización de código, la generación automatizada de pruebas, y la detección temprana de errores. Herramientas basadas en IA han contribuido significativamente a mejorar la productividad de los desarrolladores, al ofrecer sugerencias inteligentes de fragmentos de código y soluciones contextuales (Ibrahim et al., 2021), lo que permite acelerar los ciclos de desarrollo y reducir la carga cognitiva.

Asimismo, la adopción de servicios en la nube ha transformado radicalmente la forma en que se diseñan desarrolla y despliegan las aplicaciones modernas. Las arquitecturas basadas en microservicios, habilitadas por plataformas en la nube, permiten una escalabilidad flexible, resiliencia ante fallos y una gestión más eficiente de sistemas distribuidos. Esta evolución tecnológica ha facilitado la implementación de soluciones más robustas, modulares y adaptables, alineadas con las demandas actuales de disponibilidad, rendimiento y continuidad operativa.

2.3 Guía de buenas prácticas estándar PMBOK

El *PMBOK*, en su sexta edición, proporciona un marco estructurado para abordar esta complejidad. Este marco se organiza en cinco grupos de procesos fundamentales:

- **Grupo de procesos de inicio:** Proceso realizado para definir y obtener la autorización de nuevo proyecto
- **Grupo de procesos de planificación:** Se define alcance, objetivos y su curso de acción para cumplirlos.
- **Grupos de procesos de ejecución:** Proceso realizado para completar el trabajo definido a fin de satisfacer requisitos del proyecto.
- **Grupo de procesos de monitoreo y control:** Es requerido para el seguimiento, analizar y regular progreso, identificar cambios o estado de riesgos.
- **Grupo de proceso de cierre:** Proceso que conlleva la entrega del producto final y cierre del proyecto. (*PMBOK*,2017)

Para estos grupos de procesos el *PMBOK* los categoriza por Áreas de Conocimiento. Se puede definir como un área definida dentro de la dirección de los proyectos y que describe algunos procesos o tareas que lo componen. Estas áreas estas interrelacionadas, sus procesos o tareas se pueden ejecutar en paralelo con otras áreas. Las diez Áreas de Conocimiento según *PMBOK* sexta edición son las siguientes:

- **Gestión de la integración del proyecto:** Incluye los procesos y actividades que buscan combinar los grupos de procesos. El vital porque contiene tareas que orientan al alineamiento de todo el equipo de trabajo.
- **Gestión del alcance:** Incluye la sección del proyecto que ayuda a delimitar hasta donde llegara los entregables del proyecto. Importante para formar los requerimientos del proyecto.
- **Gestión del cronograma**: Incluye la administración de las tareas, recursos asignados y tiempos de entrega de estas.
- **Gestión de los costos del proyecto**: Incluye los procesos donde vemos planificar, estimar, presupuestar, gestionar y control los costos de tal manera que el proyecto cumpla con el presupuesto aprobado.
- **Gestión de la calidad del proyecto:** Todo proyecto tiene políticas o estándares de calidad por cumplir, entonces esta área busca que se cumplan la entrega de un producto de calidad.

- **Gestión de recursos del proyecto:** Incluye los procesos donde se identifican y gestionan los recursos.
- **Gestión de las comunicaciones del proyecto:** Incluye procesos donde la información llegue a los clientes en el momento adecuado.
- **Gestión de los riesgos del proyecto:** Incluye los procesos donde se mapea que puede salir mal y como se puede evitar. Respuesta ante los riesgos, control y monitoreo.
- **Gestión de las adquisiciones:** Incluye los procesos donde se necesita adquirir equipos, servicios o resultados fuera del equipo interno.
- **Gestión de los interesados del proyecto:** Acá se identifican los clientes que estarán siendo figuras importantes durante todo el proyecto, especialmente durante los requerimientos. (*PMBOK*,2017)

2.4 Agilismo y sus marcos

Desde su nacimiento en 2001 con el Manifiesto ágil, la agilidad ha pasado de ser una metodología utilizada para proyectos de software a ser una competencia buscada por las organizaciones que a su vez buscan adaptasen a un mercado cambiante. De acuerdo con, La agilidad en América latina (2021), con la crisis sanitaria del coronavirus (COVID-19) se comprobó el valor de la agilidad en la resiliencia. Las compañías tenían serios cuestionamientos por el trabajo remoto, pero por la crisis se tuvo que implementar y allí fue cuando se notó mejora de desempeño y la capacidad del trabajo a distancia, además de comodidad de no tener que trasladarse al lugar de trabajo.

El Manifiesto ágil es un documento que establece cuatro pilares claves y doce principios que guían un enfoque flexible, colaborativo y centrado en el valor para el desarrollo de sistemas y la gestión de proyectos. Los pilares agiles hacen referencia a esos 4 elementos claves para vivir el agilismo, ejemplo software funcionando, se considera con mayor valor entregar el *software* funcional por encima de la documentación detallada del sistema.

1. Individuos e iteraciones sobre los procesos y herramientas.
2. Software funcionando sobre documentación detallada.

3. Colaboración con el cliente sobre negociación contractual.
4. Respuesta antes el cambio seguir un plan.

Y los valores agiles son los valores que poseo una persona, pero estos enfocados hacia como vive una persona el agilismo en su trabajo:

1. Satisfacción del usuario.
2. Cambio a los requerimientos.
3. Liberar continuamente software funcional.
4. Trabajo conjunto entre responsables y desarrolladores.
5. Construir los proyectos alrededor de individuos motivados.
6. Conversación directa.
7. Software funcionando es el indicador de progreso.
8. Desarrollo sostenible.
9. Excelencia técnica y el buen diseño incrementa la agilidad.
10. Simplicidad.
11. Autoorganización.
12. Reflexión. (PMI, 2018)

Posterior a este auge de la agilidad, nacieron los marcos trabajo agiles, los cuales utilizan diferentes enfoques ya sea interactivo e incremental y que a cuál entregue más valor al cliente así será su elección. Dentro de estos marcos de trabajo con mayor utilización en la industria de *software* se pueden mencionar *Scrum* y *Kanban*, los cuales se abordará en las siguientes secciones.

2.4.1 Scrum

Es un marco ágil de los más populares en la actualidad, en especial cuando se habla de proyectos de software, aunque es aplicable a toda la industria.

Según la guía del *SBOK*, es marco de trabajo adaptable, iterativo, rápido, flexible y eficaz, diseñado para ofrecer un valor considerable en forma rápida a lo largo del proyecto. (ScrumStudy, 2022). Esta definición se refleja en una serie de características que se *Nota*n en equipos Scrum como lo son: la transparencia de comunicación, trabajo en equipo, responsabilidad colectica, progreso continuo, equipos empoderados, equipos multidisciplinarios y autoorganizados.

A continuación, se mencionan una seria de beneficios de porque utilizar Scrum:

1. Adaptabilidad: Los cambios en los proyectos son aceptados.
2. Transparencia: Ambiente de trabajo abierto, gracias a las diferentes ceremonias que se realizan.
3. Retroalimentación continua: A través de las ceremonias y el proceso de validar con el cliente funcionalidades.
4. Mejora continua: Los requerimientos iniciales son actualizados para refinarlos y asegurar una entrega orientada a lo que el cliente quiere realmente.
5. Entrega continua de valor: Al final cada *Sprint* (tiempo de trabajo del equipo antes de entregar al cliente, el estimada es de 2 a 4 semanas) la idea es entregar al cliente software funcional que él pueda ir validando.
6. Ritmo sostenible: Las personas involucradas pueden trabajar a este ritmo, en decir trabajan a su ritmo.
7. Entrega anticipada de alto valor: Se priorizan los requerimientos que involucran el *Minimun value product (MVP),* es decir producto mínimo viable, las funcionalidades principales que el cliente necesita.
8. Proceso de desarrollo eficiente: El *Time-Boxing (asignación de un bloque de tiempo)* y la reducción del trabajo innecesario.
9. Motivación: Las ceremonias y sus funciones conducen a que el equipo este motivado.
10. Resolución de errores de forma rápida: La colaboración y los equipos multidisciplinarios conducen a esto.
11. Entregables efectivos: El proceso de entrega continua viene de la mano de revisiones de calidad antes de entregar al cliente.
12. Centrado en el cliente: El principal valor en lo que vamos a entregar al negocio y la colaboración con ellos también, entenderles su necesidad y de qué manera lo transformamos al sistema.
13. Ambiente de alta confianza: Alineado con la motivación, las ceremonias y la colaboración como equipo generan esta característica.
14. Responsabilidad colectiva: Es el proceso de comprometer la entrega para cada sprint genera esa responsabilidad por cumplir las metas del equipo.

15. Alta velocidad: La colaboración y el equipo multidisciplinario permiten alcanzar su potencial y una alta velocidad.
16. Ambiente innovador: Las ceremonias de Retrospectiva generan un ambiente de conocimiento y lecciones aprendides entre el equipo.

Scrum tiene 19 procesos que abordan actividades del flujo del proyecto, generalmente no son secuenciales, son iterativas y se pueden solapar entre ellas ver la **Figura 2.5.** Fases y procesos *Scrum***.**

Figura 2.5. Fases y procesos *Scrum*

Fase	Procesos fundamentales de Scrum
Inicio	1. Crear la visión del proyecto 2. Identificar al Scrum Master y a los interesados del negocio 3. Formar Equipos Scrum 4. Desarrollar épicas 5. Crear el backlog priorizado del producto 6. Realizar la planificación de la liberación
Planificación y estimación	7. Crear historias de usuario 8. Estimar historias de usuario 9. Comprometer historias de usuario 10. Identificar tareas 11. Estimar tareas 12. Actualizar el backlog del sprint
Implementación	13. Crear entregables 14. Realizar el Daily Standup 15. Refinar el backlog priorizado del producto
Revisión y retrospectiva	16. Demostrar y validar el sprint 17. Retrospectiva del sprint
Liberación	18. Enviar entregables 19. Retrospectiva de la liberación

Nota: Adaptado de Procesos de Scrum (Pag. 14), por SCRUMstudy-SBOK-Guide-4th-edition-Spanish, 2022.

2.4.1.1 Artefactos

Los artefactos son herramientas utilizadas para un adecuado control del proyecto, las más populares son las siguiente:

Product Backlog: Es la lista de historias de usuarios (requerimientos) priorizadas y que además es gestionada por el PO del proyecto.

Sprint Backlog: Esta seria de lista de historias pero que pertenecen a un *Sprint*. Depende del orden del *Product Owner* se puede tener el siguiente *Sprint* casi definido, es decir a espera de la ceremonia de *planning*.

Incremento: Resultado tangible de lo realizado al finalizar un *Sprint*.

2.4.1.2 Roles

Los roles juegan un papel muy importante para entender que papel tendrá cada individuo en el proyecto. Tenemos dos:

1. Scrum Core Roles: Los comprometidos. Dentro del Scrum Core Roles vamos a tener tres personajes:
    a. *Product Owner*: La persona que trabaja representando la voz del cliente y encargado de maximizar el valor del producto. Siempre debe mantener una visión dual
    b. *Scrum Master*: Es un líder servicial y facilitador para el equipo de trabajo.
    c. *ScrumTeam*: Es el grupo de personas encargados de entender los requerimientos (historias de usuario), estimarlas y creación de entregables.
2. *Stakeholders* (clientes): Implicados. Divididos en tres personajes:
    a. Cliente: Persona u organización que va a adquirir el producto.
    b. Usuario: Las personas que serán usuarios finales del producto.
    c. Patrocinador: Persona u organización que provee los recursos para el proyecto y a quien se debe dar cuentas al final.

2.4.1.3 Ceremonias

Son una serie de reuniones llevadas a cabo con propósitos diferentes pero que agregan valor al equipo que trabaje con ellas:

Sprint Planning: Es una reunión llevada en conjunto por el PO y el SM y tiene como finalidad definir los requerimientos que se trabajaran para el próximo *Sprint*.

Daily Stand-up: Es una reunión que tarda 20 minutos y tiene como finalidad 3 preguntas muy puntuales, ¿Que hice ayer?, ¿Que voy a ejecutar el día de hoy? y ¿Tengo impedimentos?

Sprint Review: Es una reunión donde participan todos los miembros del proyecto, tiene como finalidad entregarle y presentarle al cliente la finalización de las tareas del Sprint que acaba de terminar.

Sprint Restrospective: La intención de esta reunión es revisar en conjunto con el PO, SM y equipo *Scrum* como nos fue en el Sprint que acaba de terminar. Tiene tres preguntas esenciales ¿Qué hicimos bien durante el Sprint?, ¿Qué podemos mejorar? y ¿Qué acciones vamos a tomar? Es una reflexión al trabajo recién entregado donde las lecciones aprendidas siempre se comentan.

2.4.2 Kanban

Según "La guía *Kanban* para *Scrum teams*", *Kanban* es una estrategia para optimizar el *Flow* (es el movimiento del valor a través del sistema de desarrollo de producto) de valor mediante un proceso que usa un sistema visual y con limitación del trabajo en marcha (*work-in-process*). (Vacanti y Yeret, 2021). Con esta administración del flujo, *Kanban* expone las practicas actuales para buscar mejorarlas y esta es una razón por la cual *Kanban* no tiene dos implementaciones iguales. A diferencia de *Scrum, Kanban* no es iterativo ni tiene roles específicos.

2.4.2.1 Prácticas de Kanban

Las practicas generales de *Kanban* se encuentran explicadas a continuación:

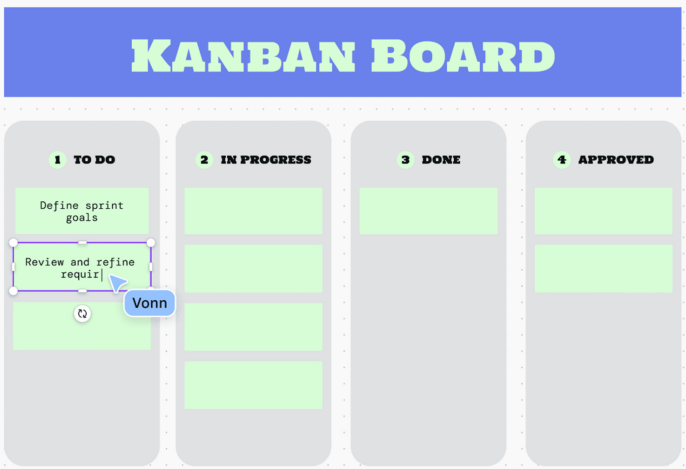
1. Visualizar el trabajo: Ver un tablero con el trabajo, riesgos y el flujo.
2. Limitar el trabajo en proceso (WIP): Limitar el trabajo en proceso para evitar acumulación, no iniciar tareas sin antes finalizar las que están en proceso.
3. Gestionar el flujo: Mantener el flujo de trabajo constante y optimizar el tiempo en que las tareas se mueven en etapas.
4. Hacer políticas explícitas: Crear las políticas que ayuden a explicar el proceso. Deben contener límites de WIP, definición de terminado, horario de reuniones, entre otros.
5. Establecer ciclos de retroalimentación: Esencial para buscar evolucionar.
6. Mejora colaborativa y evolucionar experimentalmente: Es un método de mejora de continua de procesos y esto se logra con colaboración. además, sugiere realizar experimentos donde fallar sea seguro, para mantener el cambio si fue exitoso o se pueda volver a lo anterior que no tenía problemas.

2.4.2.2 Tablero de Kanban

Como se menciona en su definición *Kanban* utiliza un sistema visual para administrar el trabajo y así expone practicas actuales para buscar mejoras. En la **Figura 2.6**. Tablero *Kanban*, se observa cómo el trabajo propuesto inicia en la columna denominada “Cola de Entrada”, donde se registran las tareas pendientes. A medida que estas avanzan en el flujo de trabajo, se trasladan progresivamente hacia la columna “Despliegue”, que representa la finalización y entrega del trabajo.

El objetivo principal de este sistema es optimizar los procesos, permitiendo detectar cuellos de botella, mejorar la gestión del tiempo y aumentar la eficiencia del equipo en la entrega de resultados. Al hacer visibles las tareas y su estado, *Kanban* fomenta la colaboración, la transparencia y la toma de decisiones informadas para lograr una mejora continua en la ejecución del trabajo.

Figura 2.6. Tablero *Kanban*



Nota: ¿Adaptado de Que es un tablero Kanban?, por página de Canva,2025, ¿Qué es un tablero Kanban? - Cómo hacerlo, ejemplos y plantillas

2.5 Herramientas tecnológicas para gestión de proyectos.

Como se ha desarrollado a lo largo de este documento, la gestión de proyectos ha experimentado una evolución significativa, influenciada por la adopción de diversas metodologías que han escalado hacia múltiples marcos de trabajo. Esta transformación ha permitido a las organizaciones adaptar sus enfoques según la naturaleza y complejidad de cada proyecto.

Es importante destacar que, dependiendo del tipo de proyecto, existen diferentes herramientas de software diseñadas para facilitar su gestión. Algunas de estas soluciones están más orientadas al ámbito tecnológico o de TI, mientras que otras se enfocan en la administración de proyectos generales dentro de las organizaciones. Esta diversidad de herramientas permite seleccionar la más adecuada según las necesidades específicas del equipo, el alcance del proyecto y los objetivos estratégicos de la empresa.

Comprender esta variedad y su aplicabilidad es clave para una gestión de proyectos eficiente, ya que permite alinear los recursos tecnológicos con las mejores prácticas metodológicas, optimizando así los resultados y la colaboración entre los distintos actores involucrados.

Pero ¿Qué hace un software de gestión de proyecto?

Un *software* de gestión de proyectos desempeña un papel fundamental en cada etapa del ciclo de vida de un proyecto. Estas herramientas se han convertido en elementos indispensables desde la fase de planificación, pasando por la ejecución, el monitoreo y hasta la documentación de lecciones aprendidas.

Entre sus principales funciones se encuentra la creación, asignación y seguimiento de tareas, lo que permite al equipo tener una visualización clara y en tiempo real de quién es responsable de cada actividad y cuáles son los plazos establecidos para su cumplimiento. Esta visibilidad facilita la coordinación de recursos, optimizando la asignación de personas, tiempo y presupuesto de manera eficiente.

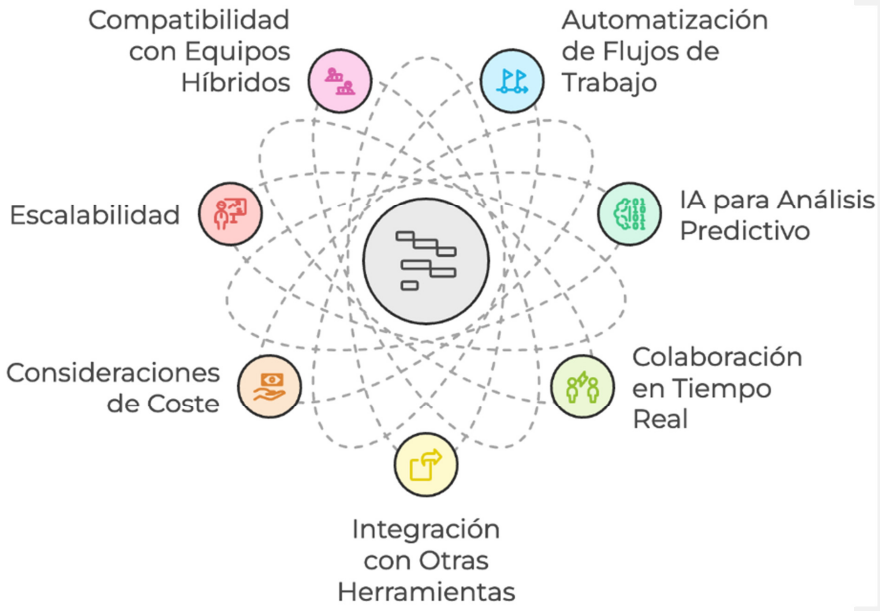
Además, estos sistemas suelen ser plataformas colaborativas, que permiten a los miembros del equipo interactuar simultáneamente, compartir actualizaciones, adjuntar documentos, comentar avances y resolver bloqueos sin necesidad de recurrir a canales externos. Esta colaboración en tiempo real mejora la comunicación, reduce errores y acelera la toma de decisiones.

En resumen, un *software* de gestión de proyectos no solo organiza el trabajo, sino que también potencia la productividad, fomenta la transparencia y fortalece el trabajo en equipo, contribuyendo directamente al éxito del proyecto.

2.5.1 Características clave de las herramientas de gestión de proyectos.

La **Figura 2.7**. Características claves en las herramientas de gestión de proyectos muestra las características que según el blog "conecta Magazine" son esenciales que deben tener una herramienta para su elección en el mercado. (Conecta Software, 2025)

Figura 2.7. Características claves en las herramientas de gestión de proyectos



Nota: Adaptada de *Características clave de las herramientas de gestión de proyectos* (Conecta Software, 2025)

Iniciando de Compatibilidad de equipos híbridos hacia la derecha una pequeña definición/ descripción de cada característica.

- Compatibilidad con equipos híbridos: Accesibilidad multiplataforma, funciones de sincronización y opciones que se adapten a equipo distribuidos.

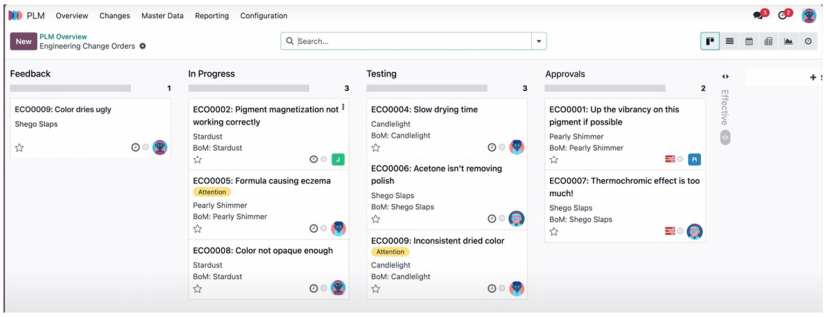
- Automatización de flujos de trabajo: Capacidad para automatizar tareas recurrentes, mejorando la eficiencia y recudiendo riesgo humano.
- IA para análisis predictivo: Tomar las consideraciones en cuanto a predicción de tareas, cargas de trabajo, plazo de tareas y permitir la automatización de asignaciones o gestionar recursos.
- Colaboración en tiempo real: Chats integrados, foros comentarios en tareas que permitan la colaboración fácilmente.
- Integración con otras herramientas: La herramienta debe poder conectarse a otros ERPs, CRM, herramientas de almacenamiento de la nube y otras aplicaciones ya utilizadas en la empresa (Microsoft teams, slack, Google workspace).
- Consideraciones de costo: Para empresas pequeñas la elección de una herramienta con menos características reducirá costos y será adaptable a su necesidad actual. Entonces todo depende de la organización existen precios y opciones que se adaptan al mercado organizaciones.
- Escalabilidad: Se puede adquirir la herramienta con menos funciones, pero adaptable si la empresa crece, entonces puedes cambiar la herramienta a mejores funciones sin cambiarla.

2.5.2 Las mejores 5 herramientas de gestión de proyectos.

Comprendiendo las características claves para un software de gestión de proyectos ahora según la revista "conecta Magazine" las mejores 10 herramientas se describen a continuación 5 de ellas, (Conecta Software, 2025):

1. **Oddo Project:** Excelente opción para empresas medianas ofreciendo un módulo robusto para gestión de proyectos. Es una solución unificada para gestionar no solo proyectos sino recursos empresariales. **Figura 2.8**. Panel Oddo donde se muestra un panel ejemplo de proyectos, cada uno muestra detalle de cantidad de tareas, fecha inicio y fin.

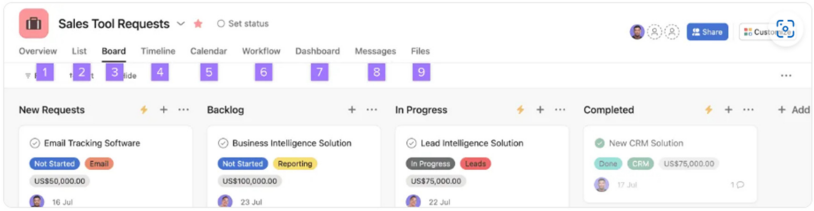
Figura 2.8. Panel Oddo



Nota: Adaptada de *PLM* (Odoo, 2025).

2. **Asana:** Cuenta con una interfaz intuitiva y flexible, se puede apreciar en la **Figura 2.9.** Panel Asana, con capacidad de adaptarse a equipos de diferentes tamaños y sectores. Ideal para proyectos de diversa complejidad.

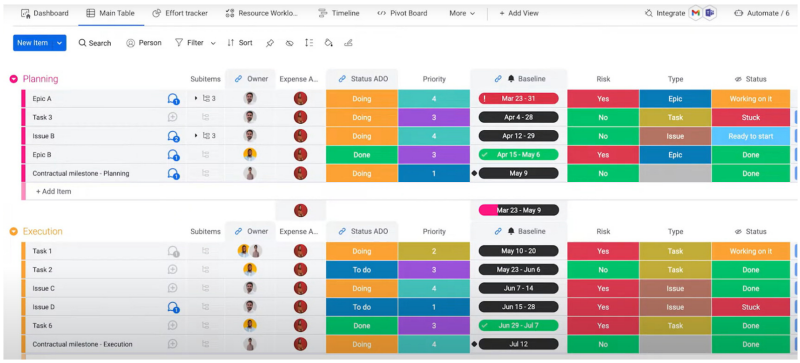
Figura 2.9. Panel Asana



Nota: Adoptada de Vistas, por página de Asana, 2025, Explora Asana

3. **Monday.com:** Es una plataforma versátil y visual que permite gestionar proyectos y tareas de manera colaborativa y eficiente. Ejemplo de un panel de proyectos en la herramienta *Monday*, **Figura 2.10**. Panel *Monday*.

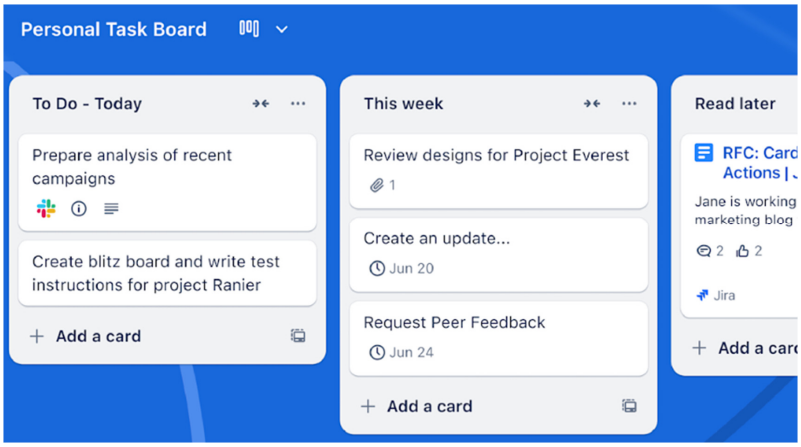
Figura 2.10. Panel *Monday*



*Nota***:** Adoptada de *Monday + Azure DevOps* (Monday.com, 2025).

4. **Trello:** Es una herramienta que presenta el tablero de la metodología *Kanban*. Es ideal para equipos pequeños o empresas que buscan simplicidad en su gestión de proyecto. Ejemplo ver **Figura 2.11**. Panel Trello.

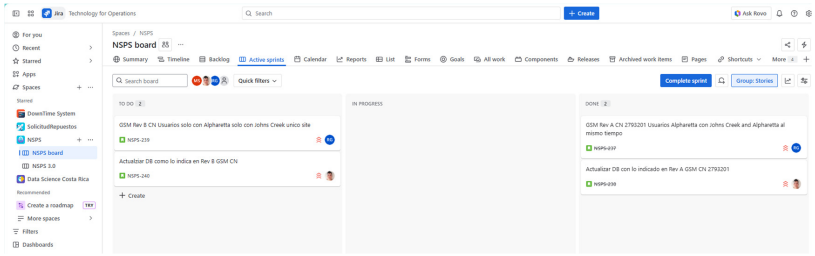
Figura 2.11. Panel Trello



Nota: Adaptada de *Boards* (Trello, 2024).

5. **Jira:** Popular en los equipos de desarrollo de *software*, pero se puede adaptar a otros proyectos. Es ideal para equipos técnicos que necesiten gestión avanzada y seguimiento de incidentes. Ejemplo ver **Figura 2.12**. Panel Jira**.**

Figura 2.12. Panel Jira



En resumen, los *softwares* de gestión de proyectos presentados ofrecen una variedad de costos y funcionalidades, lo que permite a las organizaciones seleccionar la opción más adecuada según su presupuesto, el nivel de complejidad de los proyectos y sus necesidades operativas.

Capítulo 3 Marco Metodológico

En este capítulo se expone la metodología que se utilizará para desarrollar la investigación y lograr los productos de los objetivos específicos y la solución a la problemática planteada. Como elementos del diseño metodológico de la investigación se definen las categorías y variables de estudio, además se indican las técnicas y se desarrollan los instrumentos para la recolección de datos, se identifican los sujetos y fuentes de información, se describen los productos y subproductos a obtener de cada objetivo específico y se definen los métodos y herramientas a usar en la etapa de análisis y procesamiento.

Entiéndase que para la recolección de datos deben definirse las técnicas y desarrollarse los instrumentos además de definir los sujetos y fuentes de información. Para la etapa de análisis y procesamiento de datos deben definirse los métodos y las herramientas de análisis y procesamiento que van a permitir la construcción de los entregables.

Si el estudiante lo desea puede desarrollar de una manera gráfica y en prosa las etapas que tendrá su proyecto, indicando los entregables por fase, herramientas a aplicar por fase, categorías

que se trabajan por fase, creando un esquema completo donde se podrá correlacionar fuentes, técnicas, objetivo por fase, entre otros elementos.

Puede incluir un diagrama esquemático o sinóptico que le permita al lector entender como organizó su trabajo final de graduación, de tal manera que pueda relacionar las partes del trabajo en un solo modelo. Si es un diagrama que pretende describir todo el diseño metodológico se recomienda incluirlo al final del capítulo.

3.1 Categorías de la investigación

Las categorías y variables de estudio son las características o propiedades sujetas a la observación y análisis en la investigación, y a través de su comportamiento es posible describir un fenómeno o una situación (Hernández, Hernández, C., & Baptista, 2010).

En esta sección el estudiante indicará las categorías y variables (si las hay) que trabajará en su investigación, considerando elementos de los cuales recopilará información, analizará y desarrollará soluciones. De acuerdo con el tipo de investigación y a sus objetivos debe definir una a una, las categorías de análisis y sus interrogantes o las variables con su respectiva operacionalización, en caso de usar componentes cuantitativos.

Esta fase es una de las más importantes del PFG, puesto que dicta la pauta de lo que se investigará en el campo y las formas de abordarlo en las técnicas e instrumentos seleccionados.

En cualquiera de los paradigmas de la investigación debe realizarse una matriz con la definición conceptual de categorías o variables y la forma de operacionalización.

Se recomienda plantear esta sección haciendo uso de un cuadro con al menos los siguientes elementos, pero sin olvidar el desarrollar prosa después del cuadro o antes con el fin de explicar de mejor manera al lector el contenido:

En el **Cuadro 3.1**. Categorías de la investigación se desarrollan las categorías de la investigación planteada (este cuadro puede cambiar de nombre y ampliar su estructura, si además se requiere el uso de variables, en cuyo caso se incluye su definición operacional.

Cuadro 3.1. Categorías de la investigación

Categoría: 1. Proyecto del *Downtime System*: Corresponde a las características, requisitos, ciclo de vida y los recursos para la realización del proyecto.				
Sub-categoría	**Definición conceptual**	**Pregunta generadora**	**Técnicas**	**Instrumentos**
1.1 Requisitos de usuario	Funcionalidades que espera el usuario por parte del sistema.	1- ¿Cuáles problemas buscas resolver con la implementación del proyecto? 2- ¿Cuáles son los requisitos y requerimientos del sistema? 3- ¿Cuáles son las funcionalidades prioritarias?	*Focus Group* Juicio experto. Entrevistas.	Formulario para recolección de datos. Análisis de contenido.
1.2 Ciclo de vida del proyecto.	Conjunto de etapas necesarias para completar el proyecto.	4- ¿Qué enfoque de ciclo de vida se adapta mejor a los clientes? 5- ¿Cuáles procesos involucrados en cada etapa?	Juicio experto. Filtros de idoneidad	Cuestionario filtros de idoneidad.
Categoría 2. Gestión de proyectos: Proceso mediante el cual se planifica, organiza, supervisa y controla el desarrollo de un proyecto para cumplir con el objetivo, alcance, costo y tiempo.				
Sub-categoría	**Definición conceptual**	**Pregunta generadora**	**Técnicas**	**Instrumentos**
2.1 Prácticas actuales	Conjunto de herramientas que son utilizadas hoy en día en la gestión de sus proyectos.	6- ¿Cuáles buenas prácticas actuales se pueden aplicar? 7- ¿Qué cambiaría de las prácticas actuales?	Investigación bibliográfica. Juicio experto. Entrevistas.	Ficha bibliográfica. Formulario recolección de datos.
2.2 Marcos de referencia.	Son conjuntos de buenas prácticas, enfoques, procesos y técnicas que proponen una guía para ejecutar proyectos con mayores probabilidades de éxito.	8- ¿Cuáles buenas prácticas son aplicables y ayudarían a mejorar la probabilidad de éxito del proyecto? 9- ¿Qué herramientas, técnicas y procesos se pueden utilizar en el proyecto?	Investigación bibliográfica. Juicio experto.	Ficha bibliográfica.

3.2 Población y muestra – Sujetos de investigación

El presente proyecto se realiza en *Boston Scientific* Costa Rica para la sede de Heredia específicamente involucrando los departamentos de Materiales, Industrial, Equipos y el equipo de desarrollo asignado al proyecto por parte del departamento de Tec Ops arista *Software.* Entonces corresponde a una población de 15 personas, distribuidos de la siguiente manera: una persona del Departamento de Materiales, un Industrial, tres de Equipos, cinco asignados al equipo ejecutor del proyecto y cinco *Project Mangares* de otros proyectos. Según Lohr, S. L. (3ª ed., 2022). *Sampling: Design and Analysis*, al ser una población limitada (solo los departamentos de Materiales, Equipos, Industrial y equipo de desarrollo) no es requerida una muestra.

Para realizar el estudio se agrupan los sujetos dependiendo de su área de experiencia / conocimiento de la siguiente manera: departamentos interesados (Materiales, Industrial, Equipos) y el equipo técnico asignado al proyecto por parte de Tec Ops.

Se muestra en el siguiente **Cuadro 3.2.** Sujetos de estudio la información en detalle de las razones por las cuales se consideran.

Cuadro 3.2. Sujetos de estudio

Sujeto/ Grupo de sujetos	Rol del sujeto	Razones para considerarlo en el estudio
Departamentos interesados (Materiales, Equipos, Industrial)	Los clientes con mayores conocimientos de los sistemas actuales y seleccionados para entregar requerimientos.	- Personas que conocen el negocio. - Conocen las necesidades y limitaciones actuales de los sistemas. - Pueden tomar decisiones claves en sus requerimientos. - Serán usuarios finales del nuevo sistema.
Equipo técnico asignado al proyecto	Equipo con cualidades técnicas para la construcción del software.	- Equipo encargado del desarrollo del software.
Desarrolladores con roles de	Equipo que tiene experiencia de otros proyectos en la organización.	- Equipo que ha tenido experiencias en otros proyectos.

Sujeto/ Grupo de sujetos	Rol del sujeto	Razones para considerarlo en el estudio
Project Managers en otros proyectos		

3.3 Fuentes de información

Las fuentes de información primarias se originan en la experiencia directa o en la observación sistemática. Este tipo de información se registra en materiales originales generados durante el proceso investigativo, como documentos históricos, entrevistas y encuestas. Estas fuentes son esenciales para la construcción de conocimiento riguroso, ya que proporcionan evidencia directa que sustenta el análisis y la interpretación en los estudios académicos.

Por otro lado, las fuentes secundarias son datos que no provienen de la recolección de datos, sino información que fue colectada y analizada por otros investigadores y que además esta plasmada en una publicación. Los libros, artículos e informes de investigación son de este tipo.

Para el presente proyecto se clasifican las fuentes en el **Cuadro 3.3.** Fuentes de información primarias y el **Cuadro 3.4**. Fuentes de información secundarias.

Cuadro 3.3. Fuentes de información primarias

Primaria	
Fuente	**Información por obtener**
Talleres de *focus group*	- Recopilar y priorizar los requisitos funcionales y no funcionales para el MVP, así como identificar el nivel de participación de los usuarios en el desarrollo.
Entrevista semiestructurada	**De la subcategoría:** Requisitos de usuario - Validar y definir los requisitos esenciales del MVP, y establecer la prioridad de los entregables según la perspectiva de los usuarios clave.
Filtros de idoneidad	- Identificación del ciclo de vida adecuado para el proyecto
Entrevista abierta	**De la subcategoría:** Practicas actuales - Identificación de riesgos de otros proyectos. - Lecciones aprendidas.

Primaria	
Fuente	**Información por obtener**
	- Valoración de las practicas/ herramientas actuales.

Cuadro 3.4. Fuentes de información secundarias

Secundarias	
Fuente	**Información por obtener**
Guía de los fundamentos para la dirección de proyectos (Guía del *PMBOK*)	- Identificar buenas prácticas de gestión de proyectos y seleccionar el ciclo de vida más adecuado según el estándar *PMBOK*.
Guía del *SBOK* - Guía de los fundamentos de Scrum	- Analizar las mejores prácticas de Scrum para aplicarlas en la gestión ágil del proyecto.
Guía Práctica de Ágil	- Determinar los ciclos de vida ágiles más apropiados para el desarrollo del proyecto.
La guía Kanban para Scrum *teams*	- Evaluar la implementación de Kanban para optimizar la gestión de tareas y flujos de trabajo en equipos Scrum.
Artículos de revistas o literatura en desarrollo de *software* y las mejores herramientas para la gestión.	- Recopilar conceptos clave para el marco teórico y comparar herramientas tecnológicas recomendadas para la gestión de proyectos de software.

3.4 Técnicas y herramientas para la recopilación de datos

En el presente capitulo se detallan las técnicas y herramientas necesarias para recopilar los datos de la presente investigación, así como los sujetos, el objetivo, el contexto, la técnica de aplicación y sus respectivas categorías.

3.4.1 Revisión bibliográfica y Juicio experto

Para el proyecto en estudio, se ha definido una revisión de literatura que resulta crucial, ya que actúa como guía estructural del desarrollo investigativo. Esta revisión permite identificar, analizar y sintetizar el conocimiento existente sobre el tema, lo cual facilita la formulación precisa del problema, la definición de marcos teóricos pertinentes y, posteriormente, el análisis de resultados y la propuesta de soluciones fundamentadas.

Para la revisión bibliográfica se sigue el proceso sugerido por Sánchez Huarcaya, el cual establece tres pasos para realizarla:

- Búsqueda y selección de documentos.
- Lectura y análisis de los documentos.
- Elaboración de un nuevo documento, podría ser un informe. (Sánchez Huarcaya, 2020)

El instrumento utilizado para la revisión bibliográfica se adjunta en el **Apéndice A:** Revisión bibliográfica

Preguntas generadoras que atiende: ¿Cuáles buenas prácticas actuales se pueden aplicar?, ¿Qué cambiaría de las prácticas actuales?, ¿Cuáles buenas prácticas son aplicables y ayudarían a mejorar la probabilidad de éxito del proyecto?, ¿Qué herramientas, técnicas y procesos se pueden utilizar en el proyecto?

Categoría de estudio a la que pertenece: Gestión del proyecto.

Subcategoría de estudio: Marco de trabajo y practicas actuales.

Objetivos de la revisión bibliográfica:

- Recolectar información mediante la búsqueda de fuentes confiables para el análisis de resultados y proponer soluciones.

3.4.2 Focus group

Para obtener una comprensión más profunda de los procesos actuales, es recomendable establecer reuniones semanales con el cliente. Este tipo de interacción continua no solo permite recopilar información de manera progresiva, sino que también fomenta la confianza y fortalece la relación profesional. A medida que se desarrolla esta dinámica, el cliente se siente más cómodo compartiendo detalles de sus procesos, desde los aspectos más simples hasta los más complejos.

Estas conversaciones periódicas ofrecen un espacio propicio para formular preguntas estratégicas, que pueden ayudar a anticipar escenarios complejos y explorar posibles soluciones.

El instrumento utilizado para el taller de focus *group* se adjunta en el **Apéndice D:** Talleres de focus group.

Preguntas generadoras que atiende: ¿Cuáles problemas buscas resolver con la implementación del proyecto?, ¿Cuáles son los requisitos y requerimientos del sistema? y ¿Cuáles son las funcionalidades prioritarias?

Categoría de estudio a la que pertenece: Proyecto del *Downtime System*

Subcategoría de estudio: Requisitos de usuario.

Objetivos del focus *group*/ requerimientos:

- Comprender el proceso operativo completo: Identificar y analizar las etapas del proceso operativo con el fin de determinar posibles módulos funcionales que el sistema debe incorporar.
- Mapear el flujo operativo y detectar requisitos implícitos: Reconstruir el flujo de trabajo actual, prestando especial atención a aquellos requisitos que no han sido mencionados explícitamente, pero que son esenciales para el funcionamiento del sistema.
- Explorar escenarios complejos y validar respuestas esperadas: Plantear preguntas estratégicas que permitan al cliente reflexionar sobre situaciones operativas complejas. Esto facilita la definición de cómo debería comportarse el sistema ante dichos escenarios, asegurando una cobertura funcional robusta.
- Recolectar datos relevantes vinculados al flujo operativo: Capturar información clave que, aunque no esté directamente relacionada con los requisitos iniciales, pueda influir en el diseño del sistema o enriquecer la definición de funcionalidades futuras.

Sujetos que atiende las sesiones: Existe participación de un representante por el Departamento de Ingeniería Industrial, tres por Equipos y uno por Materiales, además de que uno de los programadores. Entonces cada participante aporta desde su proceso actual y se dan comentarios entre ellos que pueden ser mejoras a sus procesos.

3.4.3 Entrevista semiestructurada

Se aplica la entrevista semi estructurada que cuenta con un guion, con el fin de validar información recibida o atender dudas de las sesiones de *focus group*. A pesar de que se cuenta con algunas preguntas, se pueden agregar o quitar otras preguntas dependiendo de las respuestas del cliente. El instrumento utilizado para la entrevista semiestructurada se adjunta en el **Apéndice B:** Entrevista semiestructurada.

Preguntas generadoras que atiende: ¿Cuáles problemas buscas resolver con la implementación del proyecto?, ¿Cuáles son los requisitos y requerimientos del sistema? y ¿Cuáles son las funcionalidades prioritarias?

Categoría de estudio a la que pertenece: Proyecto del *Downtime System*

Subcategoría de estudio: Requisitos de usuario

Objetivos de la entrevista:

- Validar información previamente recolectada para obtener mayor probabilidad de éxito en el futuro entregable del proyecto.
- Atender oportunamente las dudas del cliente es esencial para asegurar una redacción precisa y completa de los requisitos del sistema.

Sujetos a los que se les aplicara la entrevista: Esta se aplica durante las sesiones de focus *group* a los clientes, es decir, al ingeniero industrial, a los de equipos y a la persona de materiales.

3.4.4 Filtros de idoneidad y Juicio experto

En esta sección se evaluará los ciclos de vida predictivo, iterativo, incremental y ágil. Para esta sección se responderá un cuestionario en conjunto con el cliente, equipo de desarrollo y el investigador como responsable de la gestión del proyecto, esto con el fin de contar con la perspectiva de los clientes y la parte técnica del equipo. El investigador explicara cada pregunta y da contexto al cliente con ejemplos adaptados a su conocimiento. El instrumento utilizado para filtros de idoneidad se adjunta en el **Anexo 3:** Filtros de idoneidad.

Preguntas generadoras que atiende: ¿Qué enfoque de ciclo de vida se adapta mejor a los clientes? y ¿Cuáles procesos involucrados en cada etapa?

Categoría de estudio a la que pertenece: Proyecto del *Downtime System*

Subcategoría de estudio: Ciclo de vida del proyecto

Objetivos de los filtros de idoneidad:

Realizar una evaluación del tipo de ciclo de vida que se adapta mejor de acuerdo con las particularidades del proyecto.

Proceso de aplicación: Se responde a un cuestionario que conta de 12 preguntas a evaluar, se debe seleccionar cuál de las tres descripciones se adapta mejor a la situación del proyecto en particular.

Al final la recolección de datos en el cuestionario se procede a la interpretación de resultados visualizados en un gráfico de radar con tres posibles áreas (ágil, hibrido o predictivo). En este grafico los valores más cercanos al centro, significan que existe mejor ajuste a enfoques agiles y los valores más cercanos al perímetro del círculo significa que es mejor optar por un enfoque predictivo, mientras que la zona intermedia significa que el enfoque hibrido se ajustaría mejor al proyecto.

Finalmente, la decisión de que modelo utilizar se toma conjunto, se discuten resultados y se ven ventajas y desventajas de cada modelo.

3.4.5 Entrevista abierta

En este modelo, se agruparán y evaluarán las lecciones aprendidas de proyectos anteriores, con el objetivo de identificar patrones, prácticas efectivas y áreas de mejora que puedan influir en el desarrollo del proyecto actual.

El análisis de estas encuestas permitirá descubrir posibles riesgos, tanto operativos como estratégicos, que podrían afectar el cumplimiento de los objetivos. El instrumento utilizado para la entrevista abierta se adjunta en el **Apéndice D:** Entrevista abierta.

Preguntas generadoras que atiende: ¿Cuáles buenas prácticas actuales se pueden aplicar? y ¿Qué cambiaría de las prácticas actuales?

Categoría de estudio a la que pertenece: Gestión de proyectos

Subcategoría: Practicas actuales

Objetivos de la encuesta:

- Analizar los riesgos de otros proyectos para la identificación y mitigación del proyecto actual.
- Considerar si las buenas prácticas de otros proyectos pueden ser tomadas en consideración para el éxito del proyecto actual.

Proceso de aplicación:

Se seleccionarán los últimos cinco desarrolladores que han ejecutado la función de *Project Manager* para aplicarles un cuestionario con seis preguntas a evaluar con respecto al tema de lecciones aprendidas y los riegos de sus proyectos.

Se seleccionarán los últimos cinco desarrolladores que hayan desempeñado el rol de *Project Manager* en proyectos recientes, con el fin de aplicarles un cuestionario estructurado compuesto por seis preguntas clave. Este instrumento estará enfocado en evaluar sus experiencias relacionadas con las lecciones aprendidas y la gestión de riesgos durante la ejecución de sus respectivos proyectos.

El objetivo principal de esta actividad es recopilar información valiosa que permita identificar patrones comunes, prácticas exitosas, errores recurrentes y enfoques de mitigación utilizados en contextos reales. Esta retroalimentación será fundamental para enriquecer el análisis del proyecto en curso, anticipar posibles riesgos y fortalecer la planificación estratégica.

3.5 Procesamiento y productos de la investigación

En esta sección se explica cómo se obtendrá los datos y posteriormente se transformar en información necesaria para lograr el objetivo del presente proyecto.

El procesamiento de los datos recolectados en esta investigación se realiza en tres etapas secuenciales: organización y limpieza de la información, codificación y análisis temático, y síntesis de resultados para la generación de productos. Se emplean técnicas cualitativas como la codificación abierta y axial, el análisis comparativo y la identificación de categorías emergentes, complementadas con la triangulación de fuentes y métodos para robustecer la validez de los hallazgos. Los productos generados: acta de constitución, documento de requisitos, matrices evaluativas, planes subsidiarios, indicadores y tableros de control, son diseñados para facilitar la toma de decisiones, el seguimiento y la mejora continua del proyecto. Todo el proceso se realiza bajo principios éticos, garantizando la confidencialidad y el consentimiento informado de los participantes, así como la trazabilidad y documentación de cada etapa.

En base a las técnicas y herramientas la información se procesará en tres etapas explicadas a continuación:

Etapa 1: Entrevista semiestructurada y talleres *focus group*

Estas sesiones se debe recabar la mayoría de información que dará claridad a las necesidades que se buscan satisfacer al implementar el proyecto. Durante las sesiones de focus

group se busca establecer conversaciones donde el cliente cuente desde su perspectiva lo que tienen hoy sus herramientas, pero además lo que les gustaría tener con el sistema, se realiza una codificación abierta y axial. Además, como parte de los talleres se implementa una entrevista semiestructurada para conocer detalles específicos en caso de ser necesarios, se realiza una codificación de respuestas. Como resultado esta información servirá en decisión de alcance del proyecto, requisitos, módulos, prioridades y preferencia de entregables.

Etapa 2: Filtros de idoneidad, juicio experto y entrevista abierta

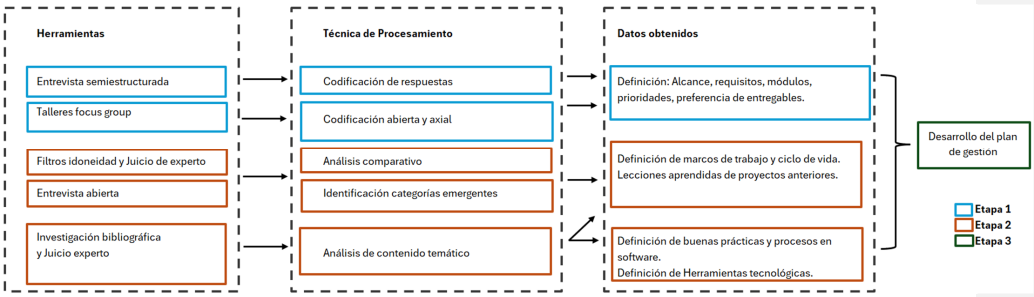
Esta segunda etapa se permite recoger la información de la herramienta de filtros de idoneidad, entrevista abierta, investigación bibliográfica y el juicio experto. Con los filtros de idoneidad y juicio experto se realiza un análisis comparativo. Con la entrevista se tiene una identificación de categorías emergentes y por últimos con la investigación bibliográfica y juicio experto, se realiza un análisis temático. El resultado será la definición de los marcos de trabajo, lecciones aprendidas, buenas prácticas y herramientas tecnológicas.

Etapa 3: Desarrollo del plan de gestión

Las salidas de las etapas 1 y 2, son insumos para el desarrollo del plan de gestión del proyecto.

Se detalla a continuación un diagrama **Figura 3.1.** Etapas investigación para entendimiento visual de las herramientas, técnicas y datos obtenidos.

Figura 3.1. Etapas investigación



3.5.1 Productos de la investigación

Objetivo específico #1: Caracterizar el proyecto "*Downtime System*", mediante el levantamiento de requerimientos funcionales y no funcionales, para la determinación de su alcance y funcionalidades esperadas.

El propósito de este objetivo específico #1 es realizar un análisis y caracterización del proyecto mediante la recopilación de los requisitos y la definición del alcance, con el fin de dar inicio formal al mismo. Como resultado de este proceso, se generarán dos entregables claves: el acta de constitución del proyecto y el documento de definición de requerimientos iniciales.

La obtención de la información necesaria se llevó a cabo mediante la aplicación de técnicas cualitativas, específicamente talleres de *focus group*, entrevistas semiestructuradas. La ejecución de estas técnicas permitió recolectar los datos esenciales para la caracterización del proyecto, lo cual constituye la base para su posterior planificación, diseño e implementación, además se pueden gestionar los planes necesarios para la gestión propuesta en el objetivo general. Se detallan a continuación ambos productos:

- Acta constitución del proyecto: Es un documento formal redactado en formato explicativo que consolida las necesidades y expectativas del cliente. Su propósito principal es establecer las bases para el desarrollo del sistema, definiendo claramente el producto mínimo viable (MVP), el alcance del proyecto, su origen, la justificación, los objetivos, el plazo estimado de entrega, el costo aproximado y los interesados clave que participarán en su ejecución. Este documento representa el punto de partida oficial del proyecto, permitiendo al equipo de trabajo alinear esfuerzos y establecer un marco común de referencia para la planificación, el diseño y la implementación de la solución propuesta.

- Documento de definición de requisitos: Es un documento redactado de forma texto explicativo donde se detallan funcionalidades del sistema, prioridad, requerimientos técnicos, restricciones y criterios de aceptación para cada requisito.

Objetivo específico #2: Identificar buenas prácticas en gestión de proyectos, mediante un estudio basado en los requisitos del *Downtime system*, para la confirmación de efectividad en el contexto del proyecto.

El objetivo consiste en consolidar una lista de buenas prácticas, procesos, artefactos, herramientas y marcos de trabajo que sean aplicables a la gestión de proyectos de desarrollo de software, específicamente en el contexto del proyecto en ejecución. Esta recopilación busca establecer una base metodológica sólida que contribuya a una gestión eficiente, estructurada y alineada con estándares reconocidos en la industria.

Para alcanzar este objetivo, se utilizarán como herramientas clave la revisión bibliográfica especializada y la realización de una entrevista abierta y filtros de idoneidad. Estas técnicas permitirán identificar, analizar y validar las prácticas más pertinentes para el entorno y éxito del proyecto. A continuación, se describen los tres productos resultantes:

- Lista de elementos aplicables: Se elaborará una lista estructurada que incluya buenas prácticas, procesos, artefactos y herramientas relevantes para el desarrollo del proyecto. Esta lista será el resultado de un análisis comparativo entre las metodologías existentes y las necesidades específicas del proyecto, con el fin de seleccionar aquellas prácticas que aporten valor y sean viables de implementar en el contexto definido.
- Matriz evaluativa de marcos de trabajo en desarrollo de software: Se presentará una tabla comparativa que analice distintos marcos de trabajo utilizados en la gestión de proyectos de *software* (como *Scrum*, *Kanban*, entre otros), considerando sus características, ventajas, limitaciones y nivel de adecuación al proyecto en cuestión. A partir de esta evaluación, se justificará la elección del marco de trabajo más apropiado para guiar la ejecución del proyecto.
- Lista de prácticas actuales: Conocer cómo trabaja el departamento actualmente para identificar con una tabla comparativa las buenas prácticas que puede adoptar el departamento.

Objetivo específico #3: Desarrollar un plan de gestión del proyecto que incluya los planes subsidiarios necesarios, para la maximización de las probabilidades de éxito y entrega oportuna del sistema.

El objetivo es la creación de los planes necesarios para la gestión y éxito del proyecto. Los insumos serán el documento de acta de constitución y el de requerimientos (producto objetivo #1) y los resultados de la investigación bibliográfica (objetivo #2). Todos los planes de gestión son explicativos y en ocasiones se implementarán tablas o cuadros para su compresión. Se detallan a continuación los planes:

- **Plan de gestión del alcance:** Establece el modo de como el alcance será definido, desarrollado, monitoreados, controlado y validado.
- **Plan de gestión de los requisitos:** Establece como se analizarán, documentan y gestionaran los requisitos.
- **Plan de gestión del cronograma:** Establece criterios y actividades a desarrollar, monitoreo y controlar cronograma.
- **Plan de gestión de costos:** Establece como se planificarán, estructuraran y controlan los costos.
- **Plan de gestión de la calidad**: Cuáles serán las políticas, metodologías y estándares de calidad que se implementarán en el proyecto.
- **Plan de gestión de los recursos:** Es una guía de cómo se deben categorizar, asignar, gestionar y liberar recursos.
- **Plan de gestión de comunicaciones:** Establece el modo se llevarán a cabo las comunicaciones del proyecto.
- **Plan de gestión de los riegos:** Establece como se estructurarán y gestionarán los riesgos.
- **Plan de gestión de las adquisiciones:** Establece como el equipo adquirirá bienes o servicios desde fuera en caso de ser necesario.

Objetivo específico #4: Establecer mecanismos de seguimiento y evaluación, mediante el uso de métricas claves que permitan tomar decisiones correctivas a tiempo durante la ejecución del proyecto.

El objetivo consiste en aplicar métricas que permitan dar trazabilidad en tiempo real al estado del proyecto, con el fin de facilitar la toma de decisiones oportunas y preventivas. Esta trazabilidad busca anticipar posibles desviaciones en el cronograma, permitiendo implementar acciones correctivas que aseguren el cumplimiento de los plazos establecidos y eviten retrasos en la entrega final.

La implementación de estas métricas será clave para monitorear el avance del proyecto de forma continua, proporcionando visibilidad sobre el desempeño de cada fase y promoviendo una gestión proactiva basada en datos. A continuación, se detallan productos a entregar:

- Reuniones de seguimiento: Establecer reuniones periódicas para revisar estado de los indicadores, validar avance y ajustar planes si fuera necesario.
- Tablero de control: Utilizar *Jira* como herramienta para visualizar en tiempo real el estado de las tareas y ayudas requeridas por el equipo.
- Establecer indicadores de desempeño: Establecer métricas para ser validadas en las reuniones de seguimiento.

3.5.2 Técnicas de procesamiento

A continuación, se describen las técnicas de procesamiento utilizadas para convertir la información obtenida en la investigación en datos relevantes para el presente proyecto.

Análisis de contenido temático: Cuando se procesa gran cantidad de información es necesario seleccionarla, analizarla, procesarla y organizarla de acuerdo con el tema de estudio. En el **Cuadro 3.5.** Contenido temático se muestra cómo se realiza el análisis de acuerdo con diferentes fuentes y que uso puede tener el proyecto.

Cuadro 3.5. Contenido temático

Fuente de información	Resumen	Uso

Codificación abierta y axial: Esta técnica permite organizar la información en categorías significativas. Primero en la codificación abierta, se deben identificar conceptos claves y asignarles un código descriptivo. Solo se van a nombrar y enlistar. Seguidamente en la axial, se debe agrupar los códigos abiertos en categorías, que permiten establecer relaciones entre los conceptos. En el **Cuadro 3.6.** Conceptos claves y **Cuadro 3.7.** Categorías se muestra cómo se realiza la codificación abierta y axial.

Cuadro 3.6. Conceptos claves

Tema mencionado	Código abierto

Cuadro 3.7. Categorías

Categorías (temática)	Códigos relacionados

Codificación de respuestas: Utilizada para identificar patrones o temas y así reducir la información. Se asignarán categorías a segmentos de texto que se consideran relevantes dentro del proyecto. En la **Cuadro 3.8.** Codificando respuestas se muestra cómo se realiza la codificación de respuestas.

Cuadro 3.8. Codificando respuestas

Sujeto	Pregunta	Tema asignado

- **Análisis comparativo:** Con el análisis comparativo se pretende establecer semejanzas y diferencias entre los datos obtenidos y plasmarlo de forma visual, para una sencilla compresión. En el **Cuadro 3.9.** Comparando marcos de trabajo se muestra cómo se realiza el análisis comparativo y el **Cuadro 3.10.** Aspectos del proyecto según idoneidad.

Cuadro 3.9. Comparando marcos de trabajo

	Scrum	**Kanban**	**Predictivo**
Características			
Ventajas			
Desventajas			

Cuadro 3.10. Aspectos del proyecto según idoneidad

Enfoque	**Aspecto**
Ágil	
Hibrido	
Predictivo	

- **Identificación de categorías emergentes:** Con esta técnica se pretende identificar temas o patrones sin imponer información. La técnica será utilizada de manera exploratoria para conocer la actualidad del departamento de software. En la **Cuadro 3.11.** Conociendo el departamento se extrae información para conocer el presente del departamento.

Cuadro 3.11. Conociendo el departamento

Fragmento	**Sujeto**

Capítulo 4 Análisis de Resultados

En el presente capítulo se exponen los resultados obtenidos a partir de la aplicación de las herramientas y técnicas descritas en el capítulo tres. Se evidencia cómo esta información contribuye al cumplimiento de los objetivos planteados para el desarrollo del proyecto. La estructura del capítulo se organiza en cuatro secciones principales:

4.1 Análisis de requisitos y generalidades del proyecto

El objetivo de esta sección es detallar los aspectos fundamentales del proyecto, abordando tanto las generalidades como el análisis de los requisitos funcionales y no funcionales que guiarán su desarrollo. A partir de sesiones de levantamiento de información con los principales clientes, se ha logrado consolidar una visión clara del alcance, ciclo de vida, cronograma, requerimientos y riesgos iniciales asociados.

La siguiente

Cuadro 4.1. Caracterización del proyecto presenta una síntesis detallada de los principales aspectos abordados durante el proceso de caracterización del proyecto. En ella se recopila y organiza la información obtenida a lo largo de las distintas sesiones de trabajo, entrevistas y análisis realizados con los actores clave. Esta información incluye tanto datos cualitativos como cuantitativos que permiten comprender el contexto actual, identificar las necesidades prioritarias, y evaluar las oportunidades y restricciones del entorno.

Cuadro 4.1. Caracterización del proyecto

Tema principal	Información recolectada	Análisis
Alcance	• La herramienta debe poder ser usada por todos las "Unidades de Negocio" de la planta de Heredia. • Su funcionalidad principal es el reporte de incidentes de líneas detenidas. • Debe cubrir la lógica de reportes que a hoy ya existen dentro de la herramienta de *Line disruption,* para lograr crear los mismos reportes.	A pesar de que los clientes actuales son del departamento de Materiales, Equipos e Industrial dado la información recolectada se pone en manifiesto que la herramienta: • Impactara a otros departamentos como *Shipping*, Bodega, Facilidades, EHS y Calidad por tanto se necesita involucrar representantes de estos departamentos.
Ciclo de vida	• Los clientes quieren entregables en periodos cortos para poder ir viendo avances y discutir posibles cambios durante la marcha.	Considerando que el cliente desea entregables en periodos cortos el ciclo de vida ágil o incremental seria la opción a elegir.
Cronograma	• Propuesta, marzo 2025. • Inicio, mayo 2025. • Plan, mayo 2025. • Puesta en marcha ejecución, junio 2025. • Implementación diciembre 2025.	Se logra definir fechas a alto nivel del proyecto considerando el esfuerzo, pero si se considera que la fecha entrega MVP es diciembre. **Ver Anexo 5:** Seguimiento proyecto Downtime System. Portafoloio Transformación Digital Heredia. Se revisa cada 15 días, para brindar actualizaciones de los proyectos. Se presento el cronograma en su momento.

Tema principal	Información recolectada	Análisis
Requerimientos	1. Sistema debe estar estructurado en seis módulos: • Módulo de gestión de usuarios. • Módulo de configuración de planta. • Módulo de registro de paro. • Módulo de soporte. • Módulo de paros de línea. • Módulo de reportes y análisis. 2. El sistema debe permitir un inicio de sesión con las credenciales de BSC sin necesidad de crear una cuenta. 3. El sistema debe permitir a un usuario administrador asignar y quitar roles a otros usuarios. 4. El sistema debe permitir a un usuario administrador brindar y quitar accesos granulares por módulo y acción a otros usuarios. 5. El sistema debe permitir consultar, ingresar, editar y eliminar líneas de producción, estaciones, procesos, áreas, equipos, *Core Teams*, modos de fallo, entre otros.	Los 36 requisitos del proyecto son la representación del MVP que debe ser entregado a diciembre. Además, se considera que los requisitos pueden ser agrupados para el inicio del proyecto valorando que daría más valor al cliente ver funcional, por ejemplo, el módulo donde se registra una línea detenida seria uno de los primeros con plan a entregar.

Tema principal	Información recolectada	Análisis
	6. El sistema debe permitir asociar líneas de producción, estaciones, procesos, áreas, equipos, estados, *Core Teams*, entre otros. 7. El sistema debe permitir clasificar los equipos en críticos o no críticos, y en activos o no activos. 8. El sistema debe permitir asignar la capacidad de cada línea (capacidad unitaria de cada estación, capacidad total, turnos de trabajo, tiempo máximo permitido de inactividad, estaciones disponibles, entre otros) 9. El sistema debe mostrar un formulario con la información respectiva para registrar un paro de línea. 10. El sistema debe permitir registrar un paro de línea. 11. El sistema debe crear también un incidente de Soporte en caso de que el paro de línea registrado sea de categoría equipos. 12. Al momento de crear un incidente de soporte, el sistema debe empezar a tomar el tiempo	

Tema principal	Información recolectada	Análisis
	activo para mantener la trazabilidad del incidente con el fin de realizar cálculos de tiempo de soporte posteriormente. 13. El sistema debe notificar automáticamente por correo electrónico a los roles específicos involucrados, al momento en que se registra un nuevo paro de línea. 14. El sistema debe calcular automáticamente, al momento de registro del paro de línea, el impacto en producción respectivo. 15. El sistema debe permitir consultar, editar y eliminar paros de línea (con control de permisos) registrados previamente. 16. El sistema debe mostrar los paros de línea registrados en formato de tabla con la información básica. 17. El sistema debe permitir consultar detalladamente la información del paro de línea registrado (id, fecha, estado, categoría, impacto en producción, incidente de soporte asociado, etc.)	

Tema principal	Información recolectada	Análisis
	18. El sistema debe recalcular automáticamente el impacto en producción en caso de modificar un paro de línea. 19. En los detalles del paro de línea, el sistema debe mostrar la comparación entre el tiempo de inactividad permitido contra el real. 20. El sistema debe permitir priorizar intervenciones de mantenimiento y soporte en base a impacto. 21. El sistema debe vincular los paros de línea con los incidentes de soporte asociados (si aplica). 22. El sistema debe permitir la búsqueda filtrada de registros de paros de línea con uno o varios parámetros (id, categoría, fecha, etc.) 23. El sistema debe permitir la exportación de datos de paros de línea en un archivo Excel. 24. El sistema debe permitir consultar y editar incidentes de equipos. 25. El sistema debe mostrar los incidentes de soporte registrados en formato de tabla con la información básica.	

Tema principal	Información recolectada	Análisis
	26. El sistema debe permitir consultar detalladamente la información del incidente de soporte registrado (id, estado, categoría, línea, estación, causa raíz, tiempos de espera y reparación, modos de fallo, paro de línea asociado, etc.) 27. El sistema debe permitir asignar y reasignar incidentes. 28. El sistema debe ser capaz de llevar trazabilidad del tiempo de los incidentes (tiempos de espera y tiempo de reparación). 29. El sistema debe permitir registrar actualizaciones y comentarios en los incidentes. 30. El sistema debe permitir realizar escalaciones manuales y automáticas en caso de exceder tiempos establecidos de incidentes 31. El sistema permitir el cierre de incidentes, actualizando el estado y el paro de línea asociado, sometiendo detalles sobre la	

Tema principal	Información recolectada	Análisis
	solución, y enviando correo electrónico a involucrados. 32. El sistema debe permitir visualizar de manera gráfica la planta con los equipos activos/en falla. 33. El sistema debe vincular los incidentes de soporte con los paros de línea asociados (si aplica). 34. El sistema debe permitir la búsqueda filtrada de incidentes con uno o varios parámetros (id, fecha, estado, equipo, etc.). 35. El sistema debe permitir la exportación de datos de incidentes en un archivo excel. 36. Reportes: • *Machine Availability Equipment Team* • *Medt time to Report (MTTR)* • *Medt time between fail (MTBF)* • *Down by Equipment/ Monthly Management Equipment Engineering.* • *Equipment Passdown* • *Unit Lost*	

Tema principal	Información recolectada	Análisis
	• *Material Dashboard*	
Riesgos iniciales identificados	• El tiempo de los recursos en el proyecto. • La adopción de la nueva herramienta en el piso de producción. • La aprobación del proyecto desde el equipo corporativo *"Architect Engagement Evaluation"* (equipo encargado de evaluar *softwares* que quieren crearse de manera local).	Si bien estos riesgos fueron identificados durante las sesiones, se debe establecer planes de acción para cada riesgo y evaluar con el equipo técnico algunos otros posibles que no se establecieron con el cliente.

4.2 Análisis del repositorio de proyectos anteriores

Con el fin de fortalecer la planificación y ejecución del proyecto *Downtime*, se ha realizado un análisis de lecciones aprendidas provenientes de iniciativas previas practicadas por el equipo de *Software*. Este ejercicio permite identificar patrones recurrentes, oportunidades de mejora y prácticas efectivas que pueden ser adaptadas al contexto actual.

Entre los hallazgos más relevantes se destacan aspectos relacionados con la comunicación con los clientes, la gestión documental, la selección de metodologías de trabajo, y el manejo de tecnologías emergentes. Estos elementos han influido directamente en el éxito o los desafíos enfrentados en proyectos anteriores, por lo que su consideración resulta clave para evitar errores repetitivos y fomentar una cultura de mejora continua.

La siguiente **Cuadro 4.2.** Análisis del repositorio de proyecto anteriores resume las lecciones aprendidas, el análisis correspondiente y las acciones concretas que se han definido para aplicar en el desarrollo del proyecto de gestión de *downtime*, asegurando una ejecución más eficiente, colaborativa y alineada con los objetivos del negocio.

Cuadro 4.2. Análisis del repositorio de proyecto anteriores

Lecciones aprendidas	Consideraciones para el presente proyecto
• Comunicación con el cliente es un tema recurrente, especialmente en cuanto a cómo manejar nuevos requerimientos y canalizar correctamente las conversaciones.	Escalar las ayudas con el patrocinador del proyecto en el momento donde los clientes no estén participando en las reuniones. Además de dejar claras las expectativas a que puede ser un cambio aceptado posterior a una entrega que cumpla con los criterios de terminado.
• Procesos internos como la documentación (repositorio, lecciones aprendidas) están ausentes, lo que puede afectar la continuidad y mejora continua.	Crear un repositorio ordenado de toda la información que estará gestionada por el plan de proyecto del *Downtime*.
• Metodologías han evolucionado de predictivo a híbrido, lo que implica cambios en la forma de trabajo.	Cada proyecto es distinto por lo tanto el departamento no puede apostar que siempre será una metodología la cual se deba seguir, por tanto, para el proyecto se evalúa el "Análisis de requisitos y generalidades del proyecto" y se aplica la plantilla de filtros de idoneidad para analizar que será lo conveniente para cumplir la fecha de entrega.
• Tecnologías desconocidas representan un reto transversal para todos los participantes.	El tema se aborda de la siguiente manera: • Tener desarrolladores con menos experiencia los cuales deben trabajar con los de mayor experiencia y brindar el apoyo necesario a los *juniors*. • Utilizar la herramienta Jira para el manejo del proyecto dado la empresa cuenta con licencias.

Lecciones aprendidas	Consideraciones para el presente proyecto
• El seguimiento de los riesgos es un proceso ausente en la mayoría de los proyectos.	• En los proyectos actualmente gestionados por el departamento, aquellos que no implican una actualización a un pilar de la planta presentan una deficiencia significativa en cuanto a la gestión de riesgos. Específicamente, se ha identificado la ausencia de una matriz de riesgos formal y de mecanismos sistemáticos de seguimiento y control. Esta carencia limita la capacidad del equipo para anticipar, evaluar y mitigar posibles impactos negativos, afectando potencialmente la eficiencia, seguridad y sostenibilidad de dichos proyectos.
• Considerar el apoyo del patrocinador de proyectos	• Establecer una comunicación constante con el patrocinador del proyecto de manera que este claro del avance y necesidades que pueda tener el proyecto.

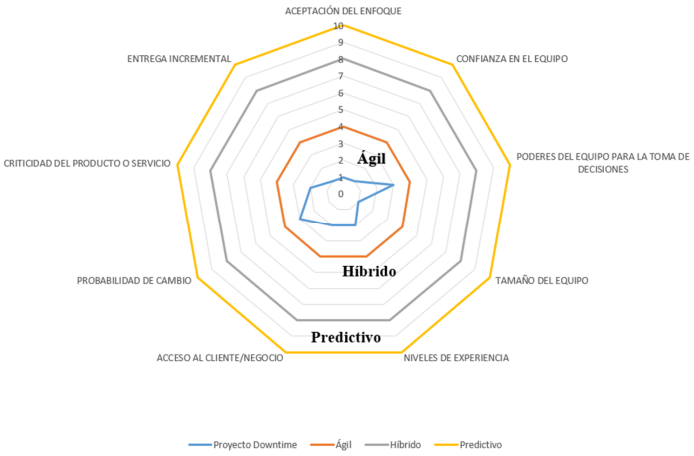
4.3 Análisis de ciclos de vida

La elección del ciclo de vida para el proyecto es un factor determinante para su éxito y su gestión, ya que define la manera en que se gestionan los requisitos, la colaboración con los clientes/ interesados, la gestión de riesgos y la entrega de valor. En el caso del proyecto *Downtime*, se realizó un análisis comparativo entre los enfoques ágil, predictivo e híbrido, considerando tanto las características identificadas durante el levantamiento de información (**Cuadro 4.3.** Análisis del ciclo de vida del proyecto) como la evidencia visual proporcionada por el gráfico radar (ver **Figura 4.1.** Gráfico radar de filtros de idoneidad).

La **Figura 4.1.** Gráfico radar de filtros de idoneidad representa el resultado del uso colaborativo de una herramienta aplicada conjuntamente por el cliente y el equipo de desarrollo. Este gráfico se construye considerando las características específicas y particularidades del proyecto, y permite visualizar de manera clara una inclinación hacia un enfoque ágil.

Dicha tendencia fue el punto de partida para la elaboración del **Cuadro 4.3.** Análisis del ciclo de vida del proyecto, cuyo propósito es profundizar en la evaluación del enfoque más adecuado para el desarrollo del proyecto, tomando como base sus condiciones contextuales, necesidades y objetivos estratégicos.

Figura 4.1. Gráfico radar de filtros de idoneidad



Cuadro 4.3. Análisis del ciclo de vida del proyecto

Características del proyecto *Downtime*	**Análisis**
Requisitos iniciales: Estos requisitos están estructurados de acuerdo con lo que hoy pueden ejecutar con sus herramientas actuales.	Los clientes mencionan que los sistemas actuales tienen deficiencias que les gustaría cambiar con el nuevo sistema, entonces se considera que los requisitos actuales tendrán cambios constantes de acuerdo con estas necesidades inciertas del cliente. Por tanto, el ciclo de vida ágil se asocia mejor a esta característica.
Colaboración con los clientes: La colaboración para entender los requisitos funcionales y valorar la definición de terminado es clave en el proyecto.	Los clientes principales (Departamento de Industrial, Materiales y Equipos) fueron asignados a brindar toda la información requerida de lo que quieren del nuevo sistema, además ellos quieren estar involucrados de cerca con el desarrollo, revisión y brindar retroalimentación de los requisitos. Además, al ser el proyecto parte del Pilar *Network* 2026, asociado a "*Digital Transformation* 2025", el patrocinador Nick quiere actualizaciones en las reuniones quincenales de este pilar. La colaboración y compromiso del cliente y patrocinador para el proyecto son altos por tanto Scrum se adapta a esto.
Gestión de los Riesgos: Los riesgos conocidos y los no conocidos necesitan la atención adecuada para no tener impacto en el proyecto.	Para esta característica el ciclo de vida hibrido se adapta mejor porque el proyecto tiene algunos riesgos previsibles porque se cuenta con experiencias en proyectos anteriores, pero podrian salir riegos emergentes que deben ser atendidos.
Integración de los sistemas: La solución integra una sola solución *Downtime* que antes eran tres	A pesar de que a hoy se tienen tres herramientas que ejecutan lo que *Downtime* viene a sustituir, el proyecto enfrenta retos y complejidad porque muchas

Características del proyecto *Downtime*	Análisis
herramientas y antes tenían limitaciones que ahora el cliente quiere mejorar.	características no son realizadas por los actuales sistemas y deben ser desarrolladas para que *Downtime* cumpla el MVP. Ejemplo: el manejo de planes de mitigación. Por tanto, esta característica se adapta mejor al agilismo.
Gestión del cronograma: Se requieren entregas de valor en tiempos cortos.	Como se mencionó anteriormente es importante entrega constante de avances del proyecto, para recibir retroalimentación y ejecutar las mejoras. Lo único confirmado es que la fase uno que sería el MVP se debe entregar a diciembre, por tanto, se puede considerar que el ciclo hibrido se puede utilizar acá.
Visualización de los requisitos funcionales: Se requiere que el cliente pueda participar activamente en sesiones donde pueda ver y validar sus requisitos.	El cliente del proyecto quiere ver los avances constantemente, quiere visualizar el valor que tendrá y buscar mejoras que puedan fortalecer la herramienta por tanto se orienta hacia el agilismo.
Entrega de valor: Deberá considerar que el cliente quiere validar sus requerimientos.	Como se ha venido mencionado el cliente quiere que los cambios y su retroalimentación sean implementados durante esta misma fase uno del sistema donde se entregará el MVP definidos por cliente y patrocinador. Se considera el ciclo ágil seria la opción, aunque puede valorarse el hibrido también porque se adapta también de alguna manera a sus características.
Impacto en la cultura organizacional: Deberá evaluarse y gestionarse la adaptabilidad de la nueva herramienta.	Un ciclo de vida ágil permite la innovación hacia la cual se dirige la organización con el plan de "*Digital Transformation* 2025".

4.4 Análisis de marco de trabajo

La selección del marco de trabajo adecuado es esencial para garantizar la eficiencia y el éxito en la gestión de proyectos. En esta sección se presenta un análisis comparativo entre los enfoques predictivo, *Scrum* y *Kanban*, considerando los principales aspectos de gestión como alcance, costo, cronograma, calidad, riesgos, métricas, roles, informes, equipo de proyecto y participación del cliente. El objetivo de la **Cuadro 4.4.** Artefactos y herramientas para gestionar el proyecto es identificar las ventajas y limitaciones de cada marco de trabajo, y así determinar cuál se adapta mejor a las necesidades y características específicas del proyecto *Downtime*.

Cuadro 4.4. Artefactos y herramientas para gestionar el proyecto

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
Alcance	EDT (Estructura de desglose de trabajo)	*Backlog*	Lista de Tareas	El proyecto *Downtime* System requiere flexibilidad para incorporar cambios frecuentes en los requisitos y una clara priorización de funcionalidades. ***Match*:** • Hace match con Scrum (backlog priorizado, historias de usuario) y Kanban (lista visual de tareas), ya que ambos permiten gestionar cambios y priorizar entregables. • El enfoque predictivo no es recomendable porque limita la adaptación a cambios.
Costo	Presupuesto definido al inicio.	Estimaciones por sprint.	Flujo continuo, control visual de trabajo en curso.	El costo debe gestionarse de forma incremental, permitiendo ajustes por sprint según el avance y cambios en el alcance. ***Match*:**

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
				• Scrum: Estimaciones por sprint y control de presupuesto iterativo. • Kanban: Permite visualizar el costo asociado al flujo de trabajo. • Predictivo: Solo útil si el alcance es fijo, lo cual no aplica aquí.
Cronograma	Fechas y entregas fijas.	*Sprints* con duración fija.	Flujo continuo, sin fechas fijas	Se recomienda definir entregas frecuentes y cortas (*sprints*), pero también mantener flexibilidad para incorporar entregas continuas si el equipo y el cliente lo requieren. ***Match*:** • Scrum: *Sprints* de duración fija, ideal para mostrar avances periódicos. • Kanban: Flujo continuo, útil si se requiere máxima flexibilidad. • Predictivo: Solo la fecha de entrega del MVP debe ser fija.

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
Calidad	Revisiones al final de cada fase	Revisiones en cada sprint (reuniones de revisión y retrospectiva)	Control de calidad en cada tarea, revisiones continuas	La calidad debe asegurarse en cada iteración, con revisiones y retroalimentación constante del cliente. ***Match*:** • Scrum: Revisión y retrospectiva en cada sprint. • Kanban: Control de calidad continuo. • Predictivo: Revisiones solo al final, lo cual es riesgoso para este proyecto.
Riesgos	Identificados y gestionados al inicio	Gestionados de forma iterativa, se revisan en cada sprint	Se gestionan conforme aparecen, enfoque en la visualización y limitación de trabajo en curso	La gestión de riesgos debe ser iterativa, revisando y actualizando riesgos en cada sprint o ciclo. ***Match*:** • Scrum: Gestión de riesgos en cada sprint. • Kanban: Visualización y gestión de riesgos conforme aparecen. • Predictivo: Solo útil para riesgos conocidos al inicio.

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
Métricas	Avance vs. plan, cumplimiento de hitos	Velocidad del equipo, *burndown chart*	*Lead time*, *cycle time*, *throughput*	Utilizar métricas ágiles como velocidad del equipo, *burndown chart*, *lead time* y *throughput* para monitorear el avance y la mejora continua. ***Match*:** • Scrum: Velocidad, burndown chart. • Kanban: *Lead time, cycle time, throughput.* • Predictivo: Avance vs. plan, menos útil en entornos cambiantes.
Roles	Gerente de proyecto, equipo funcional	*Product Owner, Scrum Master*, Equipo de desarrollo	No hay roles definidos formalmente, el equipo gestiona el flujo	Definir roles claros: *Product Owner, Scrum Master* y equipo de desarrollo, promoviendo la autoorganización y la multifuncionalidad. **Match:** • Scrum: Roles definidos y alineados con las necesidades del proyecto. • Kanban: Puede complementar, pero no define roles formales. • Predictivo: Estructura jerárquica, menos adaptable.

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
Informes	Informe de seguimiento	Informes de avance por sprint	Tablero visual actualizado, reportes de flujo	Generar informes de avance por sprint y visualizaciones en tiempo real (tableros Kanban/Jira), además de reportes mensuales para *stakeholders*. ***Match*:** • Scrum: Informes por sprint. • Kanban: Tablero visual actualizado. • Predictivo: Informes periódicos, pero menos ágiles.
Equipo de proyecto	Estructura jerárquica, roles definidos	Equipo autoorganizado y multifuncional	Equipo flexible, autoorganizado	Fomentar la autoorganización, autonomía y diversidad de habilidades en el equipo. ***Match*:** • Scrum y Kanban: Equipos autoorganizados y multifuncionales. • Predictivo: Estructura rígida, menos recomendable.
Cliente /usuario	Participa poco durante todo el ciclo de vida	Participa activamente en revisiones y priorización	Participa en la priorización y revisión del flujo de trabajo	El cliente debe participar activamente en la priorización, revisión y validación de entregables. ***Match*:**

Aspecto	Predictivo	SCRUM	KANBAN	Análisis
				• Scrum: Participación activa en revisiones y priorización. • Kanban: Participación en revisión del flujo de trabajo. • Predictivo: Participación limitada, no recomendable.

4.5 Análisis final

Como resultado del análisis integral que considera los requisitos, las generalidades del proyecto, las lecciones aprendidas de proyectos anteriores, el ciclo de vida propuesto y el marco de trabajo seleccionado, se establece la siguiente recomendación para la estructura del plan de dirección de proyecto:

- **Ciclo de vida a seleccionar:** Se definió el ciclo de vida ágil basado en iteraciones (*Sprints*), que permita flexibilidad y adaptación continua a los cambios del negocio.
- **Forma de trabajar del equipo de proyecto:** Adoptar el marco de trabajo *Scrum*, promoviendo la colaboración, transparencia y entrega incremental de valor mediante ceremonias y roles definidos.
- **Estructura del plan de gestión del proyecto:** El plan se organiza en áreas de conocimiento adaptadas al enfoque ágil:
    - **Gestión de integración:** Elaboración del Acta del proyecto que formaliza objetivos, alcance y roles. Definición de herramientas colaborativas y métricas de seguimiento.
    - **Gestión de Alcance:** Creación del *product backlog* inicial con historias de usuario priorizadas. Validación continua del alcance en cada Sprint.
    - **Gestión de cronograma:** Ceremonia de planificación para estimación de tareas y asignación de recursos. Uso de herramientas de planificación (*Jira*)
    - **Gestión de costos:** Estimación de costos por *sprint* basada en esfuerzo y recursos.
    - **Gestión de calidad:** Objetivos de calidad del proyecto, ceremonia de revisión del *sprint* para validar entregables, criterios de calidad de software.
    - **Gestión de riesgos:** Identificación y manejo de riesgos por *sprint*, registro y seguimiento de riesgos.
    - **Gestión de interesados y comunicaciones:** Identificación de interesados y participación, ceremonias regulares y herramientas de comunicaciones
    - **Gestión de cambios:** Adaptación e integración al *backlog.*
    - **Cierre del proyecto:** Retrospectiva e informe finales.

Capítulo 5 Propuesta de Solución

En este capítulo se presenta la propuesta de solución al proyecto *Downtime*, como resultado del análisis integral desarrollado en los capítulos anteriores.

La solución planteada busca responder de manera efectiva a los hallazgos obtenidos, integrando los conocimientos teóricos y prácticos adquiridos durante el proceso formativo de la maestría. Se trata de una propuesta estructurada, viable y alineada con las características y los objetivos del proyecto.

5.1 Ciclo de vida a seleccionar

Considerando el análisis del Capítulo 4 se considera que el proyecto tiene las características para ser gestionado mediante un ciclo de vida ágil con algunos elementos de la gestión predictiva, entonces se detalla a continuación en la **Figura 5.1.** Ciclo de vida ágil *Downtime* la estructura recomendada.

Figura 5.1. Ciclo de vida ágil *Downtime*



Se detalle que contiene cada etapa del ciclo de vida:

- Inicio: construcción de los elementos: Acta del proyecto, backlog inicial, identificación de partes interesadas, conformación del equipo de proyecto, roles y herramientas de control de calidad, matriz de comunicaciones
- Desarrollo: Se ejecutan las siguientes tareas por *sprint*.
    - Planificación del *sprint*: Planificar el *sprint* y estimar costos.
    - Ejecución del *sprint*: Desarrollo del *software* y pruebas por sprint
    - Cierre y lanzamiento del *sprint*: Cierra el *sprint* y valoración del incremento
    - Seguimiento y control: Control de alcance, costo, cronograma, calidad y registro y seguimiento de riesgos.
- Cierre del proyecto: Informe del cierre.

5.2 Estructura del plan de gestión del proyecto

Esta sección tiene como propósito definir cómo se desarrollará cada uno de los planes subsidiarios que conforman el plan integral de gestión del proyecto. El objetivo principal es establecer una estructura clara, ordenada y adaptable que permita una gestión eficiente de las tareas, recursos, riesgos y comunicaciones a lo largo del ciclo de vida del proyecto.

El plan de gestión se construirá de forma iterativa y colaborativa, alineado con el enfoque ágil definido en el capítulo anterior, permitiendo ajustes conforme se obtenga retroalimentación de los interesados y se avance en los entregables.

5.2.1 Gestión de integración

En el marco de trabajo *Scrum*, la gestión de la integración se enfoca en coordinar de manera efectiva todos los elementos del proyecto incluyendo personas, procesos, herramientas y entregables para asegurar que trabajen de forma alineada y sin fricciones. Esta sección del documento describe el alcance del proyecto, las herramientas colaborativas que se utilizarán y las métricas clave que permitirán monitorear el progreso y la calidad del trabajo. Todo esto se establece como un estándar de coordinación que guía la ejecución del proyecto de principio a fin, promoviendo la transparencia, la colaboración continua y la entrega de valor en cada s*print*.

Dentro del ciclo de vida definido para el proyecto, la gestión de la integración contempla tareas específicas en distintas fases clave:

- **Inicio del proyecto:** Acta del proyecto, Definición del ciclo de vida del proyecto, definición de herramientas y métricas del proyecto, Planes subsidiarios.
- **Cierre del proyecto:** Informa de la finalización del proyecto entregando el MVP definido.

5.2.1.1 Acta del proyecto

El objetivo de esta actividad es crear un documento para dar como iniciado el proyecto oficialmente y el *Product Owner* empieza a ejercer sus funciones en el proyecto. Para su realización se toma como base la información encontrada bajo el análisis del capítulo 4. El documento formal se encuentra en el Acta de Constitución que se presenta a continuación:



Boston Scientific

Downtime System Heredia

Acta de Constitución

versión 1.0

Nombre del Proyecto:

Desarrollo de un Sistema que permita reportar todo problema por línea detenida, lo que permita centralizar los problemas de down en las líneas y a su vez se autogestiona la información para reportas las unidades perdidas por línea detenida.

Equipo del Proyecto

Patrocinador: Nick O'Shea

Clientes: Departamentos de Materiales, Equipos e Industrial

Equipo de Desarrollo: 2 recursos desarrolladores de software y 1 Ingeniero de calidad de software, 1 Product Owner/ Scrum Master.

Justificación del proyecto

Este proyecto busca integrar tres sistemas que actualmente se encuentran desconectados (dos hojas de cálculo Excel y un sistema interno) y que son utilizados por distintas áreas para gestionar eventos de *downtime*, eliminando redundancias. Actualmente se utilizan los sistemas:

- VTP (departamento *Supplier Quality*): Utilizado para registrar tiempos de parada y cantidad de unidades afectadas manualmente. Temas referentes a materia primas, bodega y otros.
- *Production Tracking System (PTS)* (departamento Equipos): Sistema de incidentes para reportar fallas o paradas en los equipos utilizados en las líneas de producción.
- *Line Disruption* (departamento *Industrial Engineering*): Hoja de cálculo usada para registrar tiempos de parada por operación y estimar unidades impactadas.

Objetivos

- **Integración de sistemas**
  Unificar los tres sistemas actuales (VTP, PTS y Line Disruption) en una sola plataforma para eliminar redundancias y mejorar la eficiencia operativa
- **Centralización de información**
  Garantizar que todos los eventos de paros de línea se registren en un sistema único, accesible para los departamentos de Materiales, Equipos e Industrial.
- **Automatización y autogestión**
  Implementar funcionalidades que permitan la generación automática de alertas y el cálculo de unidades perdidas por línea detenida, reduciendo la intervención manual.
- **Mejora en la toma de decisiones**
  Proporcionar reportes dinámicos y análisis de datos que faciliten la identificación de causas raíz y la priorización de acciones correctivas

Alcance del Proyecto

El proyecto consiste en obtener los requerimientos, diseñar, desarrollar y entregar un sistema de software multiplataforma y reportes específicos que permitan a los departamentos de Materiales, Equipos e Industrial controlar y gestionar los paros de línea en la planta de Heredia. El sistema incluirá funcionalidades para registro y seguimiento de paros, alertas automáticas, generación de reportes dinámicos y análisis de datos. Además, se considera la centralización de información para los departamentos involucrados como una mejora.

Se excluyen los sites de Cartago y Coyol por el momento.

Criterios de Aceptación

Los criterios de aceptación en forma general del proyecto.

- Obtener a diciembre un software funcional que cumpla el MVP requerido.
  - Que el sistema calcule las unidades perdidas en forma automática.
  - Que el sistema tenga una interfaz amigable para la creación de los incidentes.
  - Que el sistema permita modificar la hora de cierre de tickets.
  - Que el sistema tenga su módulo de reportes.
  - Búsqueda de registros filtrados
  - Tendencias y comparativos mensuales.
  - Exportación a Excel y conexión directa con Power BI.
  - Dashboards personalizados

Costo

El presupuesto es de 75 mil dólares.

El costo en el departamento de Tec Ops es por medio de Juicio Experto: Técnica Delphi.

Acuerdos y Consideraciones

- Se tienen recursos asignados por cada departamento (Michelle, Yecson, Cristian, Cesar y Dennis) que deben atender dos reuniones semanales.
- El dueño de la aplicación para temas de auditorías será Cristian del departamento de Equipos.
- Product Owner debe asistir a la reunión Star de los proyectos en Heredia cada vez que el dueño de esta sección considere corresponde actualización del estado del proyecto para el sponsor Nick.
- El proceso de escalación siempre será con Nick.
- Soporte continuó in-house por Tec Ops.

Riesgos

- Que los clientes no asistan a las reuniones del proyecto.
- Que el cliente no entienda que algunas mejoras deben ir a fase dos.
- Que el desarrollador de tiempo medio no tenga la experiencia deseada.

Sponsor: Nick O'Shea _____

Clientes:

Materiales: Michelle Tenorio_____ Industrial: Yecson Vargas_____

Equipos: Cristian Campos _____ Equipos: Cesar Carrillo _____

Equipos: Dennis Villalobos _____

Product Owner: Mariela Bolanos _____

5.2.1.2 Definición de herramientas y métricas para el proyecto

Como parte del proceso de integración, se procede a definir las herramientas tecnológicas y las métricas que serán utilizadas para la gestión del proyecto. Esta selección se fundamenta en criterios técnicos, funcionales y de adecuación al contexto específico del equipo y del producto.

Las herramientas fueron seleccionadas en función de los roles involucrados en el desarrollo y gestión del proyecto. El equipo de desarrollo propuso aquellas soluciones de software que representan las opciones más actuales y que se ajustan de manera óptima a las necesidades técnicas del producto. Por su parte, el *Product Owner* definió los mecanismos para la gestión de tareas, priorización de requerimientos y seguimiento del avance, asegurando que las herramientas elegidas faciliten la colaboración con el cliente y la visibilidad del progreso. Se define en el **Cuadro 5.1**. Herramientas tecnológicas recomendadas, todo *software* que será apoyo en diferentes fases del ciclo de vida y además se incluye el **Cuadro 5.2.** Métricas e indicadores de control del proyecto que tienen como propósito una visualización de esas métricas que se proponen medir en los diferentes planes de gestión.

Cuadro 5.1. Herramientas tecnológicas recomendadas

Tipo	Herramientas	Uso	Justificación
Gestión de proyectos	Jira	Gestionar el *backlog*, monitoreo de progreso, visualizar gráficos, control y seguimiento de alcance.	Herramienta que la empresa ya paga y el equipo tiene licencia.
	Excel	Control y seguimiento alcance, costo, tiempo y calidad.	Es flexible para modificarse, fácil de usar y permite crear gráficos.
	Miro	Para la reunión del *sprint review*.	Ideal para dinámicas creativas, mapas mentales y permite observa patrones de pensamiento colectivo.
	Excel	Gestionar el cronograma.	Herramienta ideal para ejecutar el diseño del cronograma de forma visual.
Colaboración y comunicación	Microsof teams	Canal oficial de comunicación para reuniones y chats.	Herramienta oficial de comunicación utilizada en la empresa con licencia de Microsoft.
	Correo Electrónico	Cada sesión de Microsft teams debe tener un correo evidencia de los acuerdos de las sesiones.	Herramienta utilizada en la empresa con licencia de Microsoft y con un respaldo de 3 años en correos.
Desarrollo	Javascript, nodeJS, React, Azure	Lenguaje, librerías usadas para el sitio web, también usando los servidores Azure que la empresa requiere que usemos	Herramientas que cumplen con las políticas en tecnologías de la compañía.
Control de calidad	Cypress, X-ray	Framework de automation javascript integrable con nodeJS, X-ray plugin integrable con Jira para el manejo de *test cases.*	Herramientas que cumplen con las políticas en tecnologías de la compañía.

Cuadro 5.2. Métricas e indicadores de control del proyecto

Detalle	Métrica #	Indicador *	Herramienta^	Descripción	Valores o interpretación
General	*Story Point* (SP) #			Unidad de medida utilizada en metodologías ágiles, especialmente en Scrum, para estimar el esfuerzo relativo requerido para completar una historia de usuario o tarea. Facilita la planificación de *sprints*, el cálculo de la velocidad del equipo (cantidad de SP completados por sprint) y la proyección de entregas futuras.	Se utilizará la secuencia Fibonacci (del 1 al 20)
	Tamaño del *backlog* inicial (TBI) #			Es el tamaño total definido inicialmente de las historias de usuario y contiene los *story points.*	TBI= Sum (SP historias del backlog inicial)
	Velocity base del equipo (VB) #			Total, de SP que el equipo de desarrollo estima entregará en cada iteración.	Valor= 16 SP por *sprint*. Esto fue calculado mediante juicio experto.
		Número de iteraciones (NI) *		Cantidad de iteraciones o *sprints* que el equipo requiere para terminar el proyecto.	NI=TBI/ VB Si su resultado fuera decimal se redondea al próximo valor.
Alcance	*Story points* nuevas en el *backlog* (SPNB) #			Son historias de usuario o mejoras que el cliente menciona en las reuniones de entrega del *release*.	SPNB= sum (historias agregadas a la épica mejoras)

Detalle	Métrica #	Indicador *	Herramienta^	Descripción	Valores o interpretación
	Tamaño total del *backlog* (TTB) #			Tamaño del *backlog* actualizado con el total del proyecto.	TTB = sum (TBI+SPNB)
	Story completados por iteración (SxI) #			Corresponde a los ST que se completan en cada *sprint*.	SxI = sum (ST por iteración)
	Story por completar en el backlog (SxCB) #			Es la diferencia entre la cantidad del tamaño del *backlog* real menos los ya completados en iteraciones cerradas.	SxCB = TTB - SxI
Cronograma	Duración iteración/ sprint (DI) #			Duración en semanas de cada *sprint*.	El valor definido es de 2 semanas, obtenido por juicio experto
		Velocity del equipo (VE) *		Cantidad promedia que el equipo completa en *story points*.	VE = SP completados de los ultimos 3 *sprints* / cantidad de *sprints*
			Burndown Chart ^	Muestra el trabajo pendiente frente al tiempo restante en un *sprint* o proyecto.	Es un gráfico donde: Eje X: representa los días del sprint. Eje Y: muestra la cantidad de trabajo restante (en horas, puntos de historia o tareas). Línea ideal: indica cómo debería disminuir el trabajo si todo se ejecuta según lo planeado.

Detalle	Métrica #	Indicador *	Herramienta^	Descripción	Valores o interpretación
					Línea real: refleja el progreso real del equipo.
		Variación del *velocity* (VV) *		Analiza la consistencia del rendimiento del equipo de desarrollo a lo largo de varios *sprints*. Umbrales: Óptimo: 100% o más. Aceptable: Entre 80% y 99% No aceptable: Menos de 80%	VV = (SxI / VE) * 100 Si el resultado es mayor a 100% se está ejecutando más trabajo del planificado, por tanto, su *velocity* aumento. El menor a 100% indica una disminución del *velocity* y se completó menos trabajo del planificado. Si el valor es 100% se completó exactamente lo que se debía completar y es posible que su *velocity* se mantenga igual.
Costo		Costo por *story point* *		Se define el salario del desarrollador según categoría. * Junior * Medium * Senior	Es el costo por cada *story point*, depende del desarrollador.

Detalle	Métrica #	Indicador *	Herramienta^	Descripción	Valores o interpretación
		Total estimados (TE)*		Costo estimado story points contra el recurso	El resultado del *story points* contra el salario del recurso. • Costo indirecto es una sumatoria de cada costo de este tipo que afecto el *sprint*.
		Costo estimado por iteración (CxI) *		Costo que se espera gastar por iteración.	Suma del costo estimado en la iteración.
		Total real (TE)*		Costo real de story points completados contra el recurso	El resultado del story points contra el salario del recurso. • Costo indirecto es una sumatoria de cada costo de este tipo que afecto el *sprint*.
		Costo real iteración (CrI) *		Costo que real gastado por iteración.	Suma de todos los gastos asociados a la iteración.

Detalle	Métrica #	Indicador *	Herramienta^	Descripción	Valores o interpretación
		Variación de costo (VC) *		Diferencia entre el costo real y el estimado para cada *sprint*.	VC= CxI - CrI Si el VC es menor a cero (negativo) el sprint tiene sobrecostos. Si el VC es mayor a cero (positivo) se está gastando menos de lo planificado en el sprint. Si el VC es igual a cero, el sprint está exactamente dentro del presupuesto
Calidad		Retrabajo por *story points* con bugs (RSPB) *		Determina de manera porcentual cuantos *story points* no han cumplido con los criterios de aceptación del cliente en cada sprint. Umbral desviación máximo: 5%	RSPB = (Puntos de historia con *rework* / Total puntos de historia completados) * 100

5.2.2 Gestión de alcance

La gestión del alcance tiene como propósito principal documentar, controlar y validar todos los requerimientos iniciales y evolutivos del proyecto, asegurando que los entregables estén alineados con las expectativas del cliente y los objetivos estratégicos definidos.

Las siguientes actividades del alcance son realizadas en diferentes fases del ciclo de vida definidos:

- **Inicio del proyecto:** Definición del alcance, *Product backlog* inicial.
- **Planificación del sprint:** Planificación del *sprint*.
- **Ejecución del sprint:** Desarrollo de entregables definidos al *sprint*.
- **Cierre y lanzamiento del *sprint*:** Entrega al cliente.
- **Seguimiento y control:** Revisión y control del alcance.

5.2.2.1 Definición del alcance

La definición del alcance es el resultado del **Análisis de requisitos y generalidades del proyecto**. Esta información fue consolidada en la **sección 5.2.1.1. Acta del proyecto** durante la ejecución del plan de integración por el *Product Owner* y además se cuenta con los Requerimientos *Downtime*, con ambos documentos se afirma el delimita el alcance del proyecto.



Boston Scientific

Downtime System Heredia

Requerimientos

versión 1.0

Requerimientos

ID	Requerimiento	Tipo	Prioridad
R-01	El sistema debe estar estructurado en seis módulos: 1. Módulo de Gestión de Usuarios. 2. Módulo de Configuración de Planta. 3. Módulo de Registro de Paro. 4. Módulo de Soporte. 5. Módulo de Paros de línea. 6. Módulo de Reportes y Análisis.	Funcional	Must Have
R-02	El sistema debe permitir un inicio de sesión con las credenciales de BSC sin necesidad de crear una cuenta.	Funcional	Must Have
Módulo de Gestión de Usuarios			
R-03	El sistema debe permitir a un usuario administrador asignar y quitar roles a otros usuarios.	Funcional	Must Have
R-04	El sistema debe permitir a un usuario administrador brindar y quitar accesos granulares por módulo y acción a otros usuarios.	Funcional	Must Have
Módulo de Configuración de Planta			
R-05	El sistema debe permitir consultar, ingresar, editar y eliminar líneas de producción, estaciones,	Funcional	Must Have

	procesos, áreas, equipos, Core Teams, Modos de fallo, entre otros.		
R-06	El sistema debe permitir asociar líneas de producción, estaciones, procesos, áreas, equipos, estados, Core Teams, entre otros.	Funcional	Must Have
R-07	El sistema debe permitir clasificar los equipos en críticos o no críticos, y en activos o no activos.	Funcional	Should Have
R-08	El sistema debe permitir asignar la capacidad de cada línea (capacidad unitaria de cada estación, capacidad total, turnos de trabajo, tiempo máximo permitido de inactividad, estaciones disponibles, entre otros)	Funcional	Must Have
Módulo de Registro de Paro			
R-09	El sistema debe mostrar un formulario con la información respectiva para registrar un downtime.	Funcional	Must Have
R-10	El sistema debe permitir registrar un downtime.	Funcional	Must Have
R-11	El sistema debe crear también un incidente de Soporte en caso de que el paro de línea registrado sea de Categoría Equipos.	Funcional	Must Have
R-11	Al momento de crear un incidente de Soporte, el sistema debe empezar a tomar el tiempo activo para mantener la trazabilidad del incidente	Funcional	Must Have

	con el fin de realizar cálculos de tiempo de soporte posteriormente.		
R-12	El sistema debe notificar automáticamente por correo electrónico a los roles específicos involucrados, al momento en que se registra un nuevo downtime.	Funcional	Should Have
R-13	El sistema debe calcular automáticamente, al momento de registro del downtime, el impacto en producción respectivo.	Funcional	Must Have
Módulo de Paros de línea			
R-14	El sistema debe permitir consultar, editar y eliminar paros de línea (con control de permisos) registrados previamente.	Funcional	Must Have
R-15	El sistema debe mostrar los paros de línea registrados en formato de tabla con la información básica.	Funcional	Must Have
R-16	El sistema debe permitir consultar detalladamente la información del paro de línea registrado (Id, fecha, estado, categoría, impacto en producción, incidente de soporte asociado, etc.)	Funcional	Must Have
R-17	El sistema debe recalcular automáticamente el impacto en producción en caso de modificar un paro de línea.	Funcional	Must Have

R-18	En los detalles del paro de línea, el sistema debe mostrar la comparación entre el tiempo de inactividad permitido contra el real.	Funcional	Should Have
R-19	El sistema debe permitir priorizar intervenciones de mantenimiento y soporte en base a impacto.	Funcional	Could Have
R-20	El sistema debe vincular los paros de línea con los incidentes de soporte asociados (si aplica).	Funcional	Must Have
R-21	El sistema debe permitir la búsqueda filtrada de registros de paros de línea con uno o varios parámetros (Id, Categoría, fecha, etc.)	Funcional	Should Have
R-22	El sistema debe permitir la exportación de datos de paros de línea en un archivo Excel.	Funcional	Could Have
Módulo de Soporte			
R-23	El sistema debe permitir consultar y editar incidentes de equipos.	Funcional	Must Have
R-24	El sistema debe mostrar los incidentes de soporte registrados en formato de tabla con la información básica.	Funcional	Must Have
R-25	El sistema debe permitir consultar detalladamente la información del incidente de soporte registrado (Id, estado, categoría, línea,	Funcional	Must Have

	estación, causa raíz, tiempos de espera y reparación, modos de fallo, paro de línea asociado, etc.)		
R-26	El sistema debe permitir asignar y reasignar incidentes.	Funcional	Must Have
R-27	El sistema debe ser capaz de llevar trazabilidad del tiempo de los incidentes (tiempos de espera y tiempo de reparación).	Funcional	Must Have
R-28	El sistema debe permitir registrar actualizaciones y comentarios en los incidentes.	Funcional	Should Have
R-29	El sistema debe permitir realizar escalaciones manuales y automáticas en caso de exceder tiempos establecidos de incidentes.	Funcional	Should Have
R-30	El sistema permitir el cierre de incidentes, actualizando el estado y el paro de línea asociado, sometiendo detalles sobre la solución, y enviando correo electrónico a involucrados.	Funcional	Must Have
R-30	El sistema debe permitir visualizar de manera gráfica la planta con los equipos activos/en falla.	Funcional	Could Have
R-30	El sistema debe vincular los incidentes de soporte con los paros de línea asociados (si aplica).	Funcional	Must Have
R-31	El sistema debe permitir la búsqueda filtrada de incidentes con uno	Funcional	Must Have

	o varios parámetros (Id, fecha, estado, equipo, etc.)		
R-32	El sistema debe permitir la exportación de datos de incidentes en un archivo Excel.	Funcional	Could Have
Módulo de Reportes y Análisis			
R-33	El sistema debe mostrar Dashboards personalizados embebidos en la aplicación, desarrollados desde Power BI.	Funcional	Should Have
R-34	El sistema debe ser capaz de mostrar métricas, reportes detallados, tendencias y comparativas mensuales en base a los datos de la aplicación.	Funcional	Should Have

5.2.2.2 Product backlog inicial

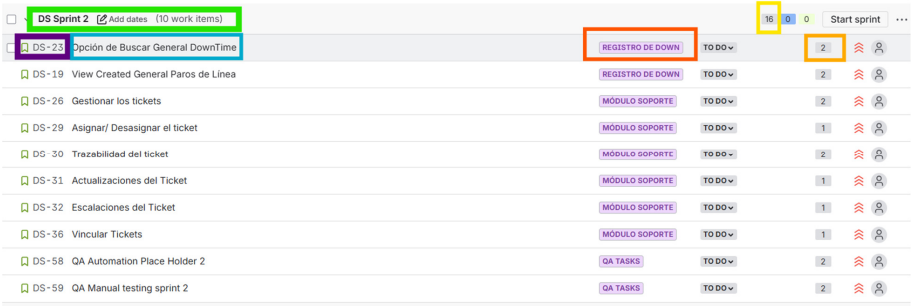
El *backlog* inicial se construye a partir del **Análisis de requisitos y generalidades del proyecto**, y constituye un insumo fundamental para la planificación ágil del desarrollo.

Este backlog representa un documento vivo, dinámico y evolutivo, que contiene una lista priorizada de requerimientos, funcionalidades, mejoras y correcciones que se desean implementar a lo largo del proyecto. Su elaboración es clave para garantizar una visión clara, compartida y alineada entre todos los actores involucrados, incluyendo el equipo de desarrollo, el *Product Owner* y los interesados clave. Su gestión es recomendable manejarla a través de Jira como se muestra en la **Figura 5.2.** Backlog inicial,

Detalles para considerar simbología y colores de la **Figura 5.2.** Backlog inicial:

- Representa ejemplo del *product baclog* que se está planeado en el *sprint*#2.
- Representa el identificador de la historia de usuario.
- Representa una descripción de la historia de usuario.
- Representa la épica que contiene la historia de usuario.
- Representa el valor *story points* que tiene la historia de usuario.
- Representa el valor total de *story points* que contiene el *sprint* en este momento.

Figura 5.2. Backlog inicial



5.2.2.3 Planificación del sprint

Esta sección tiene como finalidad definir los objetivos específicos a cumplir en cada *sprint*, partiendo del *product backlog* inicial previamente construido (**ver sección 5.2.2.2**). Dicho *backlog* contiene los requerimientos base del proyecto, organizados y priorizados, lo que permite que la planificación de los *sprints* se realice de forma estructurada, colaborativa y eficiente.

La planificación de *sprints* es una actividad clave dentro del marco ágil, ya que permite dividir el trabajo en ciclos cortos y manejables, facilitando la entrega continua de valor y la adaptación a cambios. Para lograr una planificación óptima, se deben considerar los siguientes aspectos:

5.2.2.3.1 Reunión con el team lead

La reunión entre el *Product Owner y el team lead* constituye un paso fundamental en la planificación inicial del proyecto, ya que permite alinear la visión funcional del cliente con la viabilidad técnica del equipo de desarrollo. Esta interacción se basa en el análisis del *product backlog* inicial, el cual contiene los requerimientos priorizados que servirán como base para definir el enfoque de trabajo en los primeros *sprints*.

El Product Owner tiene la responsabilidad de presentar al *team lead* los objetivos del proyecto desde la perspectiva del negocio, incluyendo las funcionalidades deseadas, las expectativas del cliente y los criterios de aceptación. Por su parte, el *team lead* aporta

su conocimiento técnico, evaluando la factibilidad de cada requerimiento, identificando posibles riesgos, dependencias y condiciones necesarias para su implementación.

Esta reunión permite:

- Validar la viabilidad técnica de los ítems del *backlog*, considerando la arquitectura, recursos disponibles y restricciones del entorno.
- Identificar ajustes necesarios en los requerimientos para que puedan ser entregables dentro de los tiempos y capacidades del equipo.
- Establecer condiciones de entrega que aseguren la calidad, estabilidad y cumplimiento de los criterios de aceptación definidos por el cliente.
- Alinear expectativas entre el negocio y el equipo técnico, evitando malentendidos y promoviendo una colaboración efectiva desde el inicio.

Además, esta instancia facilita la construcción de una estrategia de implementación progresiva, donde se priorizan los entregables de mayor valor y menor complejidad para los primeros *sprints*, permitiendo generar confianza con el cliente y obtener retroalimentación temprana.

En resumen, esta reunión no solo permite definir cómo arrancar el proyecto, sino que también establece las bases para una ejecución ágil, realista y centrada en el valor, integrando de forma efectiva la visión del cliente con las capacidades técnicas del equipo.

5.2.2.3.2 Priorización, mantenimiento y refinamiento de historias de usuario

Una vez realizada la reunión entre el *Product Owner* y *el team lead* **(ver sección 5.2.2.3.1)**, se lleva a cabo una reunión estratégica con los clientes. El objetivo principal de este encuentro es definir las expectativas de los clientes respecto a los requerimientos que deben ser abordados en los primeros *sprints*, asegurando que el desarrollo inicial se enfoque en entregar valor tangible desde las etapas tempranas del proyecto.

Durante esta reunión, el *Product Owner* presenta al cliente una propuesta de priorización basada en el *product backlog* inicial, complementada con las consideraciones técnicas aportadas por el *team lead*. Esto incluye aspectos como la viabilidad de implementación, dependencias técnicas, complejidad estimada y condiciones necesarias para cumplir con los criterios de aceptación.

Aspectos claves de la priorización y refinamiento

- **Alineación con el valor de negocio**: Se priorizan aquellas historias de usuario que generan mayor impacto en los procesos del cliente o que resuelven necesidades críticas.
- **Viabilidad técnica**: Se descartan o ajustan requerimientos que, por su complejidad o dependencia, no pueden ser abordados en los primeros *sprints*.
- **Criterios de aceptación claros**: Se refinan las historias para que cuenten con condiciones específicas que permitan validar su cumplimiento.
- **Acuerdo de *sprint***: Se establece un consenso entre el cliente y el equipo sobre los ítems que serán incluidos en el próximo sprint, considerando la capacidad del equipo y los objetivos del proyecto.

Este proceso de priorización y refinamiento permite transformar requerimientos generales en historias de usuario claras, accionables y medibles, facilitando la planificación detallada del sprint y asegurando que el equipo trabaje en lo que realmente importa para el cliente.

5.2.2.4 Desarrollo de entregables

Una vez finalizada la planificación del *sprint*, el *Product Owner* inicia el *sprint* y el equipo de desarrollo procede con la ejecución de las tareas asignadas, conforme a los acuerdos establecidos entre el *Product Owner*, el *team lead* y el cliente. Esta fase representa el núcleo operativo del proyecto, donde se transforma la planificación en resultados tangibles.

El desarrollo de entregables se basa en las historias de usuario priorizadas y refinadas, las cuales han sido validadas en términos de viabilidad técnica, valor de negocio y criterios de aceptación. Cada miembro del equipo asume responsabilidades específicas, trabajando de manera colaborativa y siguiendo los principios de transparencia, inspección y adaptación propios de los marcos ágiles.

En la **Figura 5.3.** Ejemplo *Sprint*#1 en ejecución se muestra el desarrollo del primer sprint.

Simbología y significado:

Representa las historias de usuario a ejecutar durante el *sprint*.

Representa las historias en proceso de ejecución por parte del desarrollador.

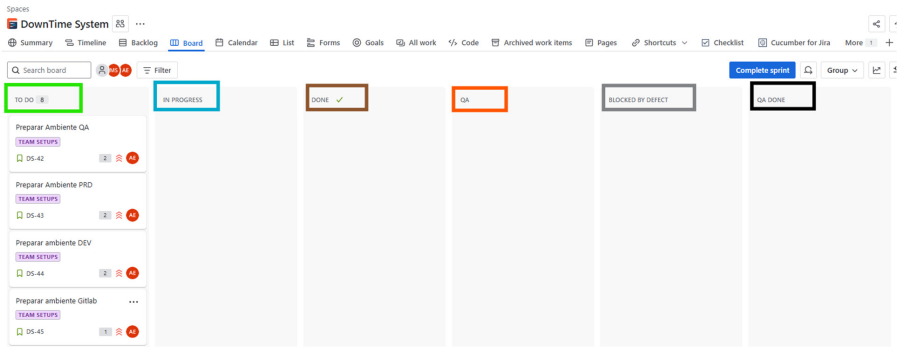
Representa las tareas finalizadas en desarrollo.

Representa las tareas listas para las pruebas de calidad de *software*.

Representa las tareas que el QA detecta con algún problema y necesitan retrabajo.

Representan las tareas que cumplen con los criterios de calidad.

Figura 5.3. Ejemplo *Sprint*#1 en ejecución



5.2.2.5 Entrega de resultado

Reunión *sprint review* con los clientes para mostrar en conjunto con el *team lead* los entregables del *sprint*. El objetivo principal es validar el trabajo realizado, generar transparencia sobre el estado del proyecto y recoger retroalimentación directa por parte del cliente o usuario final. Durante esta sesión se pueden generar historias nuevas de usuario que serán agregadas al *backlog* en su épica “Mejoras”.

5.2.2.6 Herramientas, técnicas y artefactos

Durante esta sección se detallan las herramientas, técnica y artefactos utilizados durante el alcance del proyecto:

- ***Product backlog***: Como se mencionó anteriormente este contiene las historias de usuario ya transformadas en tareas priorizadas que ejecutará el equipo técnico de desarrollo, además estarán divididas en *sprints*. Para la gestión del *backlog* se utilizará la herramienta Jira como se muestra en la **Figura 5.4.** *Downtime product backlog*.

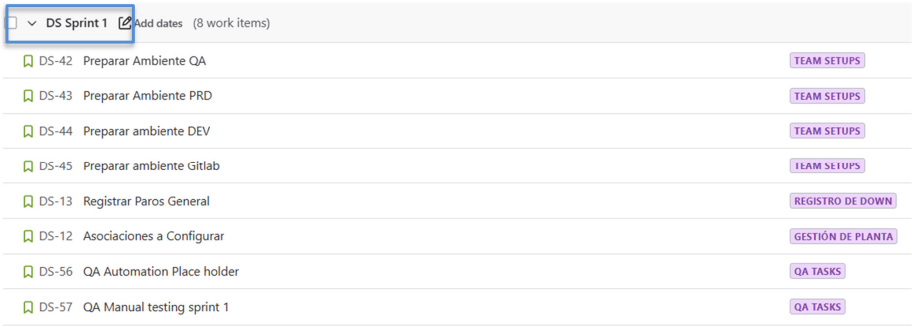
Figura 5.4. *Downtime product backlog*



Type	Key	Summary	Status	Comments	Sprint
	DS-2	Gestión de Usuarios	TO DO	Add comment	
	DS-1	Asignar o Quitar Roles	TO DO	Add comment	DS Sprint 7
	DS-3	Brindar o Quitar accesos granulares	TO DO	Add comment	DS Sprint 7
	DS-5	Gestión de Planta	TO DO	Add comment	
	DS-12	Asociaciones a Configurar	TO DO	Add comment	DS Sprint 1
	DS-4	Configuración Lineas de Producción	TO DO	Add comment	DS Sprint 7
	DS-6	Configuración de Estaciones	TO DO	Add comment	DS Sprint 8
	DS-7	Configuracion de Procesos	TO DO	Add comment	DS Sprint 8
	DS-8	Configuración de áreas	TO DO	Add comment	DS Sprint 8
	DS-9	Configuración de Unidad de Negocio	TO DO	Add comment	DS Sprint 8

- ***Sprint backlog*:** Es un subconjunto de tareas que el equipo de desarrollo se compromete a finalizar durante un sprint. Igual que el *Product backlog*, el *sprint backlog* se gestionará en Jira. En la **Figura 5.5**. *Sprint* #1 *Downtime* se puede ver un ejemplo del *sprint backlog*.

Figura 5.5. *Sprint* #1 *Downtime*



DS Sprint 1 Add dates (8 work items)

DS-42	Preparar Ambiente QA	TEAM SETUPS
DS-43	Preparar Ambiente PRD	TEAM SETUPS
DS-44	Preparar ambiente DEV	TEAM SETUPS
DS-45	Preparar ambiente Gitlab	TEAM SETUPS
DS-13	Registrar Paros General	REGISTRO DE DOWN
DS-12	Asociaciones a Configurar	GESTIÓN DE PLANTA
DS-56	QA Automation Place holder	QA TASKS
DS-57	QA Manual testing sprint 1	QA TASKS

- **Incremento:** Es la suma de todos los elementos del *product backlog* completados durante el *sprint* y los anteriores, que cumplen con la definición de terminado.

- ***Burnup chart***: Grafico que se generar desde Jira que evalúa los *story points* completados en el *sprint* contra lo comprometido por el equipo de desarrollo.

Análisis del *velocity* del equipo y control del alcance

El análisis del *velocity* del equipo de desarrollo es una práctica esencial dentro de la planificación ágil, ya que permite validar la capacidad real del equipo para ejecutar tareas dentro de un sprint, en función de su rendimiento histórico y su disponibilidad actual.

Para el proceso de análisis planteado para el proyecto se establecen los siguientes datos bases que ayudaran a construir métricas que generen control y seguimiento en el proyecto.

- **Tamaño del *backlog inicial* (TBI):** Es el tamaño total definido inicialmente de las historias de usuario y contiene los *story points*.
- ***Velocity* base del equipo (VB):** Total de SP que el equipo de desarrollo estima entregará en cada iteración.
- **Número de iteraciones (NI):** Cantidad de iteraciones o *sprints* que el equipo requiere para terminar el proyecto.

A partir de estos datos base se pueden establecer las herramientas para el control y generar métricas que ayuden a visualizar el estado del proyecto.

- ***Story points* nuevas en el backlog** (SPNB): Son historias de usuario o mejoras que el cliente menciona en las reuniones del *sprint review*.
- **Tamaño total del *backlog* (TTB):** Tamaño del *backlog* actualizado con el total del proyecto.
- ***Story* completados por iteración (SxI):** Corresponde a los ST que se completan en cada *sprint*.
- ***Story* por completar en el *backlog* (SxCB):** Es la diferencia entre la cantidad del tamaño del backlog real menos los ya completados en iteraciones cerradas.
- ***Burnup chart***: Grafico que se generar desde Jira que evalúa los *story points* completados en el *sprint* contra lo comprometido por el equipo de desarrollo.

Se explica seguidamente en el **Cuadro 5.3.** Métricas gestión alcance como estas métricas son utilizadas y ayudan a monitorear y ser preventivos antes situaciones durante el proyecto.

Cuadro 5.3. Métricas gestión alcance

Métrica	Punto control	Utilización
Story points nuevas en el backlog (SPNB)	Durante la planificación del siguiente *sprint* se evalúan las historias nuevas agregadas al *backlog*. El umbral máximo permitido es de 48 SP que estarán en la épica "Mejoras".	Conocer si el proyecto se está saliendo de la fecha de entrega prevista, para tomar medidas preventivas que ayuden a corregir el estado actual.
Story por completar en el *backlog* (SxCB)	Al final el *sprint* se actualiza el dato considerando que es la diferencia entre la cantidad del tamaño del *backlog* real menos los ya los completados en iteraciones cerradas.	Esto nos ayuda a monitorear el progreso del proyecto.
Burnup chart	Se genera al cerrar el *sprint* y se revisará durante la retrospectiva del *sprint*.	Este grafico evalúa la capacidad de entrega del equipo, confirmando si el equipo está finalizando lo comprometido o se ejecutar alguna medida adicional.
Burndown chart	Se revisa en los *dailys* para monitorear el avance del equipo en el proyecto.	Este grafico ayuda a identificar tareas que pueden tener impedimentos y que se necesite ayuda extra para seguir su flujo.

Para el adecuado seguimiento de estas métricas, al final de cada sprint se debe completar el **Cuadro 5.4.** Plantilla de Seguimiento y Control.

Cuadro 5.4. Plantilla de Seguimiento y Control

Nombre del Proyecto	Downtime System
Costo	₡28,350,000.00
Tamaño del backlog inicial (TBI) en SP	100
Velocity base del equipo (VB)	16
Número de iteraciones (NI) *	9
Duración iteración/ sprint (DI) #	2 semanas
Costo estimado por iteración (CxI) *	₡3,150,000.00

	Tabla de control alcance, costo y cronograma												
#	**Métrica o Indicador**	***Sprint #1***	***Sprint #2***	***Sprint #3***	***Sprint #4***	***Sprint #5***	***Sprint #6***	***Sprint #7***	***Sprint #8***	***Sprint #9***	***Sprint #10***	***Sprint #11***	***Sprint #12***
1	Duración del *sprint*												
2	*Velocity* del equipo (VE)												
3	*Story* completados por iteración (SxI)												
4	Variación del *velocity* (VV)												
5	Costo estimado por iteración (CxI) *												
6	Costo real iteración (CrI) *												
7	*Story Points* nuevas en el *backlog* (SPNB) #												
8	Tamaño total del *backlog* (TTB) #												
9	*Story* por completar en el backlog (SxCB)#												
10	Variación de costo (VC) * por iteración												

	Tabla de control alcance, costo y cronograma												
#	**Métrica o Indicador**	***Sprint #1***	***Sprint #2***	***Sprint #3***	***Sprint #4***	***Sprint #5***	***Sprint #6***	***Sprint #7***	***Sprint #8***	***Sprint #9***	***Sprint #10***	***Sprint #11***	***Sprint #12***
	Variación de costo												
	Tabla control de calidad												
11	Cobertura pruebas automatizadas												
12	Cobertura pruebas manuales												
13	*Story point* con *bugs*												

5.2.3 Gestión de cronograma

En la siguiente sección se procede a detallar como se realizó la estimación de tiempo inicial, cuales herramientas y artefactos se utilizaron para gestionar el tiempo y así como métricas de control y seguimiento.

Actividades relacionadas con la gestión del cronograma dentro del ciclo de vida del proyecto:

- **Inicio del proyecto:** Estimación del tiempo inicial.
- **Planificación del sprint:** Refinamiento del *backlog*, estimación de tareas.
- **Cierre y lanzamiento del sprint:** Ajuste y revisión del *sprint* y calendario.
- **Seguimiento y control:** Revisión del *burndwon chart*, revisar *velocity* del equipo.

5.2.3.1 Estimación del tiempo inicial

Para la siguiente estimación se basa en los datos de la definición de requisitos y el *project charter*.

- El desarrollo del software se realiza en *sprints* de cada dos semanas.
- El equipo del proyecto será: 1 desarrollador tiempo completo y 1 desarrollador medio tiempo, 1 ingeniero de calidad de *software* en medio tiempo y 1 *Product Owner* tiempo completo.
- El *velocity* del equipo es de 16 s*tory points* por *sprint* de 2 semanas.
- El rol del *Scrum Master* será asumido también por el *Product Owner*.
- El MVP del proyecto debe entregarse el 1 de diciembre 2025.
- La fecha de inicio del proyecto fue: 16 de junio 2025

Después de revisar esta información, se proceder a realizar la estimación en cuanto a que el proyecto debe quedar finalizado el 1 de diciembre.

Lo primero es que el *backlog* inicial está muy completo con respecto a los requisitos iniciales (se pueden esperar mejoras que se van a ir incluyendo en una épica de "Mejoras" que se van a analizar para incluirlas durante este MVP a entregar). En conjunto *Product Owner* y equipo de desarrollo estiman los *story points* en las historias de usuario.

Al realizar esta primera estimación por parte de los desarrolladores, se determina que el backlog consta de 144 *story points*, y se estima que se necesitan nueve *sprints* para terminar el proyecto el 20 de octubre, pero se considera que el tiempo faltante para el 1 de diciembre será

utilizado para toda entrada nueva al *backlog* en la épica “Mejoras” considerando las lecciones aprendidas de otros proyectos.

Las historias de usuario son agrupadas por *sprint* (recordar siempre están estimadas) para que los recursos asignados siempre tengan tareas a ejecutar, asegurando el uso eficiente de los recursos. Además, para la estimación se priorizan las tareas del *backlog* de tal manera que las que generen más valor al cliente sean las primeras en entregarse. En el

Cuadro 5.6. Tareas y estimación se puede observar la lista completa de tareas estimadas y agrupadas por sprint, así como la distribución de *story points* por *sprint* en el **Cuadro 5.5.** Distribución de *Sprint* y *story points*..

Cuadro 5.5. Distribución de *Sprint* y *story points*

Sprint	Story point
Sprint#1	15
Sprint#2	16
Sprint#3	16
Sprint#4	17
Sprint#5	16
Sprint#6	16
Sprint#7	16
Sprint#8	15
Sprint#9	9

Cuadro 5.6. Tareas y estimación

Sprint	Req. #	Historia de usuario	Story points
Sprint#1	DS-42	Preparar Ambiente QA	2
	DS-43	Preparar Ambiente PRD	2
	DS-44	Preparar ambiente DEV	2
	DS-45	Preparar ambiente Gitlab	1
	DS-13	Registrar Paros General	2
	DS-12	Asociaciones a Configurar	2
	DS-56	QA Automation Place holder	2
	DS-57	QA Manual testing sprint 1	2
	Total Story Points		**15**
Sprint#2	DS-23	Opción de Buscar General DownTime	2
	DS-19	View Created General Paros de Línea	2
	DS-26	Gestionar los tickets	2
	DS-29	Asignar/ Desasignar el ticket	1
	DS-30	Trazabilidad del ticket	2
	DS-31	Actualizaciones del Ticket	1
	DS-32	Escalaciones del Ticket	1
	DS-36	Vincular Tickets	1
	DS-58	QA Automation Place Holder 2	2
	DS-59	QA Manual testing sprint 2	2
	Total Story Points		**16**
Sprint#3	DS-33	Cierre del ticket	1
	DS-22	Automatización Impacto Línea	3
	DS-15	Notificaciones de Tickets	2
	DS-14	Crear Ticket Clasificación Equipo	2
	DS-20	Priorizacion del Ticket Para Equipos	1
	DS-27	View Created tickets de Equipo	1
	DS-28	Detalle del ticket Equipos	2
	DS-60	QA Automation Place Holder 3	2
	DS-61	QA Manual testing sprint 3	2
	Total Story Points		**16**

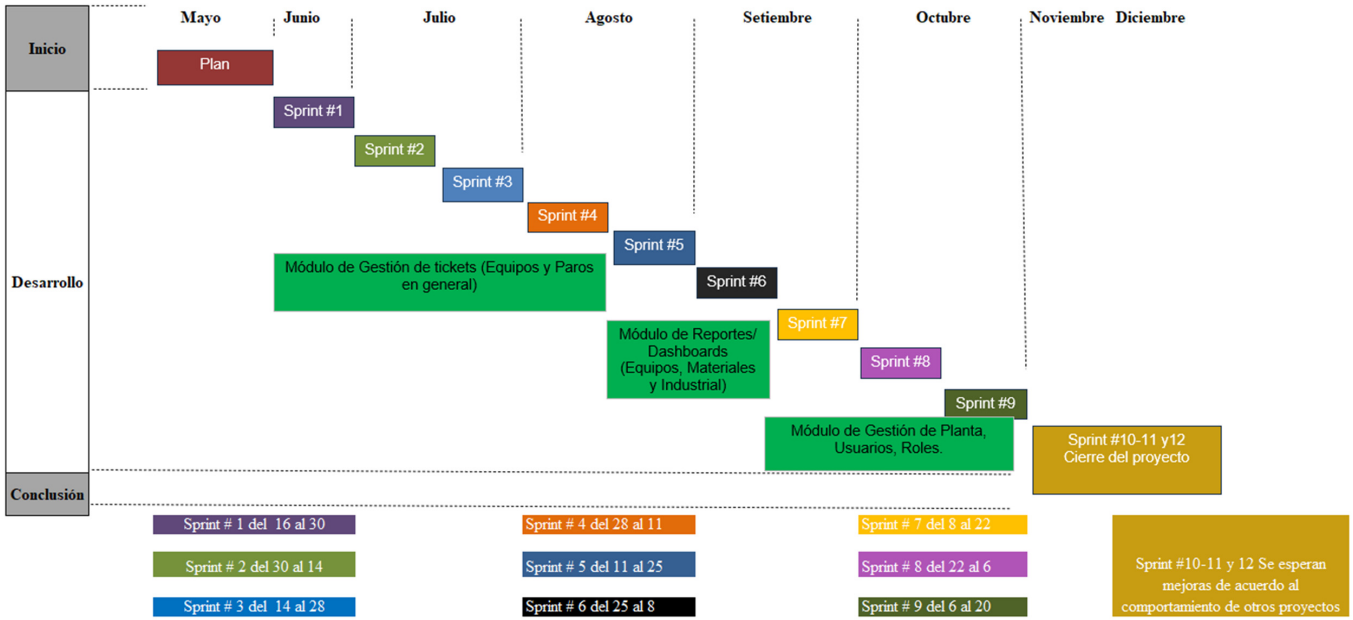
Sprint	Req. #	Historia de usuario	Story points
Sprint#4	DS-18	Gestión/ Cierre Paros de Línea	3
	DS-40	Dashboards	3
	DS-38	Exportar Datos	3
	DS-34	Servicio Correo para cierre del ticket	2
	DS-24	Exportar Datos	2
	DS-62	QA Automation Place Holder 4	2
	DS-63	QA Manual testing sprint 4	2
	Total Story Points		**17**
Sprint#5	DS-41	Métricas/ Reportes	5
	DS-49	Medt time to Report (MTTR)	3
	DS-50	Medt time between fail (MTBF)	3
	DS-54	Reportes de Michelle	2
	DS-64	QA Automation Place Holder 5	2
	DS-65	QA Manual testing sprint 5	2
	Total Story Points		**17**
Sprint#6	DS-53	Unit Lost	2
	DS-55	Machine Availability Jec Trend	3
	DS-52	Equipment Passdown/ revisarlo	3
	DS-51	Down by Equipment/ Monthly Management Equipment Engineering	2
	DS-48	Machine Availability Equipment Team	2
	DS-69	QA Automation Place Holder 6	2
	DS-70	QA Manual testing sprint 6	2
	Total Story Points		**16**
Sprint#7	DS-21	Asociación Ticket con Paros	2
	DS-16	Automatizar Calcular Time	2
	DS-35	Visualizar Equipos Activos/ Falla	2
	DS-1	Asignar o Quitar Roles	2
	DS-3	Brindar o Quitar accesos granulares	1
	DS-4	Configuración Lineas de Producción	3
	DS-71	QA Automation Place Holder 7	2
	DS-72	QA Manual testing sprint 7	2
	Total Story Points		**16**

Sprint	Req. #	Historia de usuario	Story points
Sprint#8	DS-6	Configuración de Estaciones	2
	DS-7	Configuracion de Procesos	3
	DS-8	Configuración de áreas	2
	DS-9	Configuración de Unidad de Negocio	2
	DS-10	Configuración de Core Teams	2
	DS-73	QA Manual testing sprint 8	2
	DS-74	QA Automation place holder 8	2
	Total Story Points		**15**
Sprint#9	DS-47	Configurar PRIME CODE	3
	DS-11	Configuración Modos de fallos	2
	DS-75	QA Manual testing sprint 9	2
	DS-76	QA Automation place holder 9	2
	Total Story Points		**9**
Sprint# 10-11 y 12		Se esperan mejoras. Acción preventiva.	

Como se ha mencionado anteriormente si bien en el *backlog* inicial está muy completo, el refinamiento y estimación real va a ir ocurriendo durante la planificación del *sprint* en conjunto equipo de desarrollo, cliente y *Product Owner* para tener mayor precisión de la estimación del tiempo, costo y realizar ajustes validando el actual cronograma.

En la **Figura 5.6.** Cronograma de desarrollo, se puede apreciar el calendario a alto nivel que se puede desarrollar con el *backlog* inicial del proyecto, en él se puede destacar los 9 *sprints* y la consideración de los *sprints* #10-11 y 12 como recomendación de lecciones aprendidas de otros proyectos donde se ubicaran las nuevas historias de usuario de la épica de "Mejoras" para un total de 24 semanas.

Figura 5.6. Cronograma de desarrollo



5.2.3.2 Herramientas, técnicas y artefactos

Para la gestión del tiempo se consideran las siguientes herramientas, técnicas y artefactos a ser utilizados:

- ***Story Points* y Juicio experto**: La técnica de juicio experto se basa en consultar a personas con experiencia relevante para estimar duración, esfuerzo, costo o recurso de las historias de usuario y de esta manera se asigna los puntos de historia en cada tarea. Para el proyecto se considera validar *story points* y horas trabajadas de acuerdo con la siguiente **Tabla 5.1.** *Downtime story points***.**

Tabla 5.1. *Downtime story points*

Story point	**Horas trabajadas**
1	8
2	16
3	24
5	32

Durante el ejercicio de la estimación de tareas el *Product Owner* enuncio la historia y ambos desarrolladores analizan la necesidad del cliente y criterios de aceptación para finalmente indicar los *story points* considerados de acuerdo con la **Tabla 5.1.** *Downtime story points***.**

- ***Product backlog*:** Todos los requisitos iniciales y cualquiera adicional deben incluirse, este incluye, descripción, criterios de aceptación, prioridad, subtareas, *story point* y recurso. El *backlog* como se mencionó anteriormente en se gestionará mediante la herramienta *Jira*.
- ***Burndown chart*:** Es un gráfico generado por *Jira* conforme va avanzando el proyecto. Este gráfico permite conocer el progreso realizado en cada *sprint*, mostrando a cantidad de trabajo restante en *story points* y lo contrasta con el tiempo

que resta del *sprint* para ayudar a predecir si se cumplirá con el tiempo y el alcance del *sprint* y realizar ajustes de ser necesario.

- ***Timeboxing*:** La técnica es utilizada para definir y limitar los tiempos de actividades y ceremonias lo cual ayuda a ser más eficientes con el tiempo durante los *sprints*. Se detallan las actividades y ceremonias del proyecto:
    - Duración del *sprint*: Dos semanas (10 días hábiles).
    - *Sprint planning*: 1 hora de planeamiento.
    - *Daily stand-up*: Los martes y jueves de 20 minutos.
    - *Sprint review*: 1 hora por *sprint*.
    - *Sprint retrospective*: 1 hora por *sprint*.
    - Refinamiento del *backlog*: 3 horas.

Se muestra en el siguiente **Cuadro 5.7.** Fechas claves *Downtime* funcionalidades claves y fecha estimada de entrega de cada uno de estos entregables.

Cuadro 5.7. Fechas claves *Downtime*

Release	Funcionalidades	Entrega estimada
***Sprint* # 1**	Registrar paros en general	30 de junio
***Sprint* # 2**	Gestión de *tickets* Asignar desasignar *tickets* Vistas de *tickets* Actualización de *tickets* Escalaciones de *tickets*	14 de julio
***Sprint* # 3**	Automatización de impacto Priorización de *tickets* Notificaciones de *tickets* Cierres de *tickets* Crear *tickets* clasificación de equipos	28 de julio
***Sprint* # 4**	Gestión y cierre paros de línea *Dashboards* Exportar datos de las vistas	11 de agosto
***Sprint* # 5**	Reportes:	25 de agosto

Release	Funcionalidades	Entrega estimada
	Reportes de equipos Reportes de Materiales	
***Sprint* # 6**	Reportes: Reportes de equipos Reportes de industrial	8 de setiembre
***Sprint* # 7**	Gestión de usuarios y roles. Automatizar los tiempos entre incidentes	22 de setiembre
***Sprint* # 8**	Gestión de planta	6 de octubre
***Sprint* # 9**	Gestión de planta	20 de octubre
***Sprint* # 10-11-12**	Mejoras	1 de diciembre

5.2.3.3 Control del cronograma

Durante el control del cronograma se utilizarán las métricas que fueron definidas en la **sección 5.3.1.2** Definición de herramientas y métricas para el proyecto. A continuación, se amplía un poco más de ellas.

Las métricas ayudarán a establecer cuantitativamente si el proyecto está sufriendo cambios de fechas y poder actuar preventivamente en caso de ser necesario, ahora si más bien es positivo el cambio, la parte de agradecimiento al equipo es también importante para hacerlos sentir valorados. Para medir el cronograma se establecieron las siguientes métricas o indicadores:

- **Duración iteración (DI):** Este tiempo fue establecido considerando que los clientes quieres ver avances recurrentemente y con el equipo de desarrollo. Se estableció que será cada dos semanas las entregas al cliente.

Después del inicio y durante el proyecto se podrán obtener las siguientes métricas que interesan para valorar el desempeño del equipo.

- ***Velocity* del equipo (VE):** Es la cantidad promedio de *story points* que el equipo completa en cada iteración.
- **Variación del *velocity* (VV):** Analiza el rendimiento del equipo, comparando el *velocity* del equipo con la cantidad de SP completados.

- ***Burndown chart***: Ayudará al *Product Owner* a visualizar si el equipo terminará el trabajo planificado dentro del sprint definido.

Se explica seguidamente en el **Cuadro 5.8.** Control de cronograma como estas métricas son utilizadas y ayudan a monitorear y ser preventivos antes situaciones durante el proyecto.

Cuadro 5.8. Control de cronograma

Métrica	Punto control	Utilización
***Velocity* del equipo (VE)**	Al finalizar el sprint se debe revisar si el equipo cumplió con los objetivos propuestos en el tiempo establecido (Duración de la iteración (DI) de acuerdo con el *velocity* estimado del equipo.	Se podría estimar cuando se puede acabar el proyecto. Se puede valorar los primeros *sprints* contra las historias base del proyecto y proyectar la finalización de este.
Variación del *velocity* (VV)	Al final el *sprint*, durante el *sprint review*, se compara el *velocity* del equipo (VE) contra la cantidad de *story points* completados, para si determinar si se está cumpliendo los umbrales definidos.	Esta métrica se debe revisar cada *sprint* pero considerar que si en dos *sprints* consecutivos su *velocity* a cambiado, se recomienda ajustar el *velocity* del equipo y revisar las estimaciones de fin del proyecto. Buscar causas y posibles soluciones de ser necesario.
Burndown chart	Se revisa en los *dailys* para monitorear el avance del equipo en el proyecto.	Este grafico ayuda a identificar tareas que pueden tener impedimentos y que se necesite ayuda extra para seguir su flujo.

Para el adecuado seguimiento de estas métricas de tiempo, al final de cada sprint se debe completar la plantilla del **Cuadro 5.4.** Plantilla de Seguimiento y Control.

5.2.4 Gestión de costos

Aun cuando dentro de la empresa *Boston Scientific* este tipo de proyectos no tienen en sus tareas temas de presupuesto o costo se presenta a continuación como puede ser validado el costo

que tendrá este proyecto con fines de conocimiento y posible utilidad de la información en un futuro.

Actividades relacionadas con la gestión de costos dentro del ciclo de vida del proyecto:

- **Planificación del sprint:** Estimación de costos por sprint.
- **Seguimiento y control:** Control de costos.

5.2.4.1 Estimación de costos por sprint

La estimación de costos por *sprint* constituye una parte fundamental del plan de gestión de costos del proyecto *Downtime*, ya que permite proyectar y controlar los recursos financieros necesarios para cada iteración del desarrollo. Esta estimación se basa en el enfoque ágil de trabajo incremental, donde cada *sprint* representa un ciclo de entrega de valor medible, el **Cuadro 5.9.** Estimación costo por *sprint* y recursos se recomienda para calcular los costos.

Considerando la importancia de costos se recomienda considerar la estimación de *story points* del *backlog* para proyectar el costo de cada *sprint* dependiendo de:

- *Story points* asignados al recurso y su nivel de puesto.
- Costos indirectos: Licencias de *software*, infraestructura tecnológica, *Product Owner*.

Cuadro 5.9. Estimación costo por *sprint* y recursos

# *Sprint*	Recurso	Nivel del puesto	*Story points estimados*	Horas	Total Estimado
***Sprint* # 1 al 9 *****Sprints*#10,11 y 12**	Andres Gutiérrez	Senior	8	40	₡2,000,000.00
	Jesus Fuentes	Junior	4	20	₡500,000.00
	Allen Monge	Quality Software II	4	-	₡500,000.00
	Costos indirectos Licencias y azure	-	-	-	₡150,000.00
	Costo estimado por iteración (CxI)			-	₡3,150,000.00
	Total, por 9 *sprints*			-	₡25,350,000.00

**Nota*: Costo no permitido de visualizar

***Nota*: Estos *sprints* son generados para la épica "Mejoras" del proyecto como parte de las lecciones aprendidas de otros proyectos.

5.2.4.2 Control de costos

Partiendo de la estimación de costos, es fundamental implementar un control de estos en cada *sprint*. Este control debe permitir identificar desviaciones respecto a lo planificado y valorar oportunamente cualquier incremento en los costos del proyecto.

Durante el control del costo se utilizarán las métricas que fueron definidas en la **sección 5.2.1.2** Definición de herramientas y métricas para el proyecto. A continuación, se amplía un poco más de ellas.

Las métricas seleccionadas permitirán establecer de manera cuantitativa si el proyecto experimenta variaciones respecto al costo estimado inicialmente. Además, el uso de estas métricas

contribuirá a la construcción de un factor histórico de información, el cual servirá como referencia valiosa para la estimación y gestión de costos en futuros proyectos de software.

Para medir el costo se establecieron las siguientes métricas o indicadores:

- **Costo estimado por iteración (CxI):** Representa el monto proyectado de inversión para cada sprint a lo largo del ciclo de vida del proyecto.
- **Costo real iteración (CrI):** Corresponde al gasto efectivamente incurrido durante el desarrollo de cada sprint. El análisis del costo real permite validar la precisión de las estimaciones iniciales y detectar posibles desviaciones.
- **Variación del costo (VC):** Es la diferencia entre el costo estimado y el costo real de cada iteración.

Se explica seguidamente en el **Cuadro 5.10.** Control de costos como estas métricas son utilizadas y ayudan a monitorear y ser preventivos antes situaciones durante el proyecto.

Cuadro 5.10. Control de costos

Métrica	Punto control	Utilización
Costo real por iteración (CrI)	Al finalizar el sprint se debe revisar si el equipo cumplió con los objetivos propuestos (*story points* completados)	Se podría determinar cuánto ha sido el costo real del *sprint.*
Variación del costo (VV)	Al final el *sprint*, como tarea del *Product Owner*, se compara la diferencia entre el costo real y el estimado para cada *sprint*.	La métrica de variación del costo permitirá identificar de manera cuantitativa si el proyecto está experimentando sobrecostos, ahorros o si el gasto real se mantiene alineado con lo planificado

Para el adecuado seguimiento de estas métricas de costos, al final de cada sprint se debe completar la plantilla del **Cuadro 5.4.** Plantilla de Seguimiento y Control.

5.2.5 Gestión de calidad

La gestión de la calidad en el proyecto *Downtime* se fundamenta en asegurar que los entregables cumplan con los requisitos funcionales, técnicos y operativos definidos por los usuarios finales y los estándares corporativos

Actividades relacionadas con la gestión de la calidad dentro del ciclo de vida del proyecto:

- **Inicio:** Roles y responsabilidades, herramientas de control de calidad.
- **Planificación del *sprint*:** Definición de criterios de aceptación y done.
- **Ejecución del *sprint*:** Aseguramiento de calidad mediante las pruebas automatizadas.
- **Cierre y lanzamiento del *sprint*:** Recopilación de la retroalimentación de interesados (ceremonias de *review* y retrospectiva).
- **Seguimiento y control:** Pruebas de usuario y plantilla de seguimiento y control.

5.2.5.1 Roles y responsabilidades

El equipo de trabajo cumple un papel fundamental en el aseguramiento de la calidad y para ellos se establecen ciertos roles:

- **Desarrolladores:** Primero son los desarrolladores que tienen a su cargo implementar software de calidad y realizar sus propias pruebas, corregir defectos hasta asegurar cumplan los criterios de aceptación del cliente.
- **Desarrollador QA:** Encargados de revisar el código del desarrollador intentando retar al sistema con pruebas automatizadas y manuales.
- ***Product Owner*:** Debe especificar de forma clara y conciso las historias de usuario, entendiendo los criterios de aceptación, revisar los entregables, valida cumplimiento de los criterios y realiza pruebas con el usuario del cual también reciba el *feedback*.

5.2.5.2 Criterios de calidad de software

Los criterios de calidad son parámetros fundamentales que garantizan que el producto cumpla con las expectativas del cliente y los estándares definidos. Estos criterios se establecen en conjunto con el cliente y representan el mínimo indispensable que debe cumplir el software para ser considerado aceptable. Además, sirven como base para la definición de "**Done**" y para las pruebas de aceptación.

Es importante destacar que los criterios de aceptación se encuentran definidos dentro de cada historia de usuario. Estos criterios son revisados y validados por el QA durante la fase de pruebas, lo que permite verificar, de manera individual, que cada historia cumpla con los requisitos establecidos antes de ser aprobada. Ejemplo **Cuadro 5.11**. Criterios calidad *Downtime* generales.

Cuadro 5.11. Criterios calidad *Downtime* generales

Categoría	Criterio de Calidad
Funcionalidad	El sistema debe cumplir con todos los requerimientos funcionales definidos en el backlog.
Usabilidad	La interfaz debe ser intuitiva, accesible y fácil de usar para el usuario final.
Rendimiento	El software debe responder en menos de 2 segundos para operaciones críticas.
Seguridad	El sistema debe proteger los datos sensibles mediante autenticación y cifrado.
Compatibilidad	Debe funcionar correctamente en los navegadores definidos.
Mantenibilidad	El código debe seguir estándares de desarrollo que faciliten su mantenimiento y evolución.
Escalabilidad	La solución debe poder adaptarse al crecimiento de usuarios o datos sin pérdida de rendimiento.
Fiabilidad	El sistema debe tener una tasa de error menor al 1% en producción.
Cumplimiento	Debe cumplir con normativas legales o estándares específicos dentro de Boston Scientific.

5.2.5.3 Herramientas, técnicas y artefactos

Para la gestión de la calidad se consideran las siguientes herramientas, técnicas y artefactos a ser utilizados:

5.2.5.3.1 Estrategias de calidad

Se detalla las herramientas definidas dentro de la **sección 5.2.1.2** Definición de herramientas y métricas para el proyecto.

Cuadro 5.12. Métodos para validar calidad de software

Estrategia	Descripción	Acción	Herramientas
Pruebas de QA	El desarrollador de QA desarrolla un plan de pruebas para requisitos funcionales y no funcionales.	Las pruebas serán ejecutas durante el ciclo de vida del proyecto y entre cada iteración se deben realizar pruebas de integración, funcionales, rendimiento y calidad de código.	Cypress, X-ray
Pruebas de usuario	El usuario final deberá probar las funcionalidades entregadas por cada *sprint*, validando cumpla criterios de aceptación y sus expectativas.	Al final cada *sprint* el usuario realizará sus pruebas y durante el siguiente sprint se reunirá con el *Product Owner* para la retroalimentación del sprint anterior.	N/A
Criterios de aceptación	Cada historia de usuario debe estar clara, con los objetivos y expectativas para el desarrollo.	En la reunión del refinamiento debe revisarse cumpla con los criterios de aceptación claros.	Jira
DoD	Criterios que el equipo considera que una historia de usuario debe tener para considerarse completa y lista para ser entregada.	Para ser aceptada debe cumplir las siguientes condiciones: • El código ha sido implementado y cumple con los criterios de aceptación definidos. • La historia ha sido validada por QA y no presenta defectos abiertos. • El código ha sido revisado y aprobado por al menos otro miembro del equipo.	Jira

Estrategia	Descripción	Acción	Herramientas
		• Se ha validado que la interfaz cumple con los lineamientos de diseño establecidos. • El *Product Owner* ha revisado y aceptado la historia como completada.	

5.2.5.3.2 Ceremonia de revisión del sprint

Como parte integral del proceso de calidad del proyecto, se establece por el *Product Owner* la ceremonia de revisión del *sprint* al finalizar cada iteración. Esta ceremonia tiene como propósito principal inspeccionar el trabajo completado, validar entregables con los clientes y recibir retroalimentación directa que permita ajustar el rumbo del desarrollo.

Se plantea la siguiente plantilla **Cuadro 5.13.** Control de revisión *sprints* para el manejo de la revisión del *sprint*:

- **Fecha**: El día de la ceremonia del *sprint review*.
- ***Sprint Review* #**: Se indica el número de *sprint* el cual se está validando.
- **Objetivos del *sprint***: Las expectativas que tiene el cliente con el *sprint* de acuerdo con las tareas que debían ejecutarse dentro del *sprint*.
- **# Historia/Tarea Jira**: Este dato viene del número de historia indicado en el *sprint backlog* de *Jira* en el proyecto.
- **Descripción:** Detalle del # historia de usuario previamente definido en Jira.
- **Estado:** Si la historia fue completada Si o No de acuerdo con los criterios de aceptación.
- ***Feedback*/ Épica de Mejoras**: Todo comentario que los clientes valoren les falto incluir dentro de los criterios de aceptación será tomado como mejora y se debe incluir en la épica de "Mejoras" luego de esta reunión por el *Product Owner*.

Cuadro 5.13. Control de revisión *sprints*

Fecha: junio 30, 2025
Sprint r*eview* # 1
Objetivos del *sprint*:
Prepara ambientes para la publicación del sistema y base de datos para registrar paros y asociaciones necesarias.

Entregables			
# Historia/Tarea Jira	**Descripción**	**Estado**	***Feedback*/ Épica Mejoras**
DS-45	Preparar ambiente Gitlab	Aprobado	-
DS-42	Preparar Ambiente QA	Aprobado	-
DS-43	Preparar Ambiente PRD	Aprobado	-
DS-44	Preparar ambiente DEV	Aprobado	-
DS-13	Registrar Paros General	Aprobado	-
DS-12	Asociaciones por Configurar	Aprobado	-
DS-56	QA Automation Place holder	Aprobado	-
DS-57	QA Manual testing sprint 1	Aprobado	-

5.2.5.3.3 Ceremonia de retrospectiva del sprint

La retrospectiva representa un mecanismo de retroalimentación interna que promueve la autorregulación del equipo, el aprendizaje organizacional y la adaptación progresiva. En este contexto, resulta fundamental gestionar el componente humano del equipo, explorando en profundidad cómo se sintieron los integrantes durante el sprint.

Utilizando las herramientas de la empresa Boston Scientific se recomienda la utilización de la herramienta digital *Miro* como parte del proceso de mejora continua en la ceremonia de retrospectiva del *sprint* que es organizada por el *Product Owner*, siguiendo un patrón estructurado que permita maximizar su valor colaborativo y analítico.

El patrón propuesto para su implementación contempla las siguientes fases:

- **Preparación del Tablero:** Se debe configurar un tablero específico para la retrospectiva, incorporando secciones como "¿Qué salió bien?", "¿Qué se puede mejorar?", "¿Qué acciones proponemos?", y "¿Cómo nos sentimos?". Esta estructura permite abordar tanto aspectos técnicos como emocionales del sprint.
- **Recopilación de percepciones:** Los participantes deben registrar sus observaciones de manera individual, utilizando *Nota*s adhesivas digitales, íconos o emojis para expresar sus ideas y emociones. Esta fase puede realizarse de forma anónima para fomentar la sinceridad y reducir sesgos sociales.
- **Agrupación y análisis colaborativo:** Una vez recopiladas las entradas, el equipo procede a agrupar temas similares, identificar patrones y priorizar los aspectos más relevantes mediante votaciones o discusiones guiadas. Esta etapa promueve el pensamiento crítico colectivo y la toma de decisiones informada.
- **Definición de acciones de mejora:** A partir del análisis, se establecen acciones concretas que serán incorporadas en el siguiente sprint. Miro permite vincular estas acciones con responsables y fechas, facilitando el seguimiento posterior.
- **Documentación y trazabilidad:** El tablero generado será ligado al Jira del proyecto, asegurando la trazabilidad de las decisiones tomadas y su integración en el ciclo de desarrollo.

La información generada durante las retrospectivas de *sprint* debe ser documentada de manera clara y estructurada para facilitar el seguimiento de acciones, la mejora continua del equipo

y la trazabilidad de decisiones. Para ello, se recomienda utilizar la plantilla del **Cuadro 5.14.** Plantilla *Sprint Retrospective*.

Cuadro 5.14. Plantilla *Sprint Retrospective*.

Sprint #	Link Miro	Qué salió bien	¿Qué podría mejorar?	Métricas del Sprint	Comentarios adicionales

Control de calidad

Se amplía la métrica descrita en la **sección 5.3.1.2** Definición herramientas y métricas del proyecto.

- **Retrabajo por *story points* con *bugs* (RSPB):** Dentro de la revisión de calidad se pueden encontrar detalles que pueden llevar el retrabajo por parte del desarrollador.

Se explica seguidamente en el **Cuadro 5.15.** Control de calidad como estas métricas son utilizadas y ayudan a monitorear y ser preventivos antes situaciones durante el proyecto.

Cuadro 5.15. Control de calidad

Métrica o indicador	Punto de control	Uso de la información brindad por la métrica
Retrabajo por *story points* con *bugs* (RSPB)	Al finalizar el sprint se actualiza el indicador validando si el desarrollador termino o si debe ser una tarea para terminar en el segundo *sprint*.	El indicador ayuda a monitorear si el calendario debe cambiar y la calidad de tareas desarrolladas.

Para el adecuado seguimiento de estas métricas de calidad, al final de cada sprint se debe completar la plantilla del **Cuadro 5.4.** Plantilla de Seguimiento y Control.

5.2.6 Gestión de interesados y comunicaciones

Durante este enfoque ágil esta sección se caracteriza por su enfoque iterativo, transparente y colaborativo, promoviendo la participación activa y continua de los interesados clave a lo largo del ciclo de desarrollo.

Actividades relacionadas con la gestión de los interesados y comunicaciones dentro del ciclo de vida del proyecto:

- **Inicio del proyecto:** Identificar interesados, definir las reuniones y matriz de comunicación.
- **Ejecución del sprint:** Comunicación del estado del proyecto y reporte de avances.
- **Cierre y lanzamiento del sprint:** Ajustes de reuniones.

5.2.6.1 Identificación de interesados y participación

A lo largo del ciclo de vida del proyecto, es fundamental garantizar la participación activa del cliente, los interesados clave y, en este caso, del patrocinador del proyecto. Esta colaboración continua es esencial para asegurar que la entrega final del proyecto corresponda al MVP definido en el acta de constitución del proyecto, cumpliendo con los requisitos y expectativas iniciales.

Con el fin de optimizar la gestión estratégica de los interesados, se propone la implementación de la **Cuadro 5.16.** Matriz de interesados, la cual permitirá capturar de manera sistemática y estructurada la información relevante de cada parte interesada. Esta herramienta será clave para facilitar la toma de decisiones, establecer canales de comunicación efectivos y asegurar que todos los involucrados se alineen con los objetivos del proyecto en cada una de sus fases.

Cuadro 5.16. Matriz de interesados

Nombre del interesado	Rol	Expectativas	Estrategia de comunicación	Frecuencia de comunicación	Responsable de la comunicación
O'Shea, Nick	Patrocinador	Cierre del MVP diciembre 2025	Presentación del updated en reunión "Star"	Mensual	*Product Owner*
Tenorio Villalobos, Michelle	Cliente	Calidad del producto y funcionalidad completa	Presentaciones de avance y demostraciones, reunión aclaración de requerimientos.	Semanal y Quincenal	*Product Owner*

Nombre del interesado	Rol	Expectativas	Estrategia de comunicación	Frecuencia de comunicación	Responsable de la comunicación
Villalobos Zuniga, Dennis	Cliente	Calidad del producto y funcionalidad completa	Presentaciones de avance y demostraciones, reunión aclaración de requerimientos.	Semanal y Quincenal	*Product Owner*
Vargas Solis, Yecson	Cliente	Calidad del producto y funcionalidad completa	Presentaciones de avance y demostraciones, reunión aclaración de requerimientos.	Semanal y Quincenal	*Product Owner*

Nombre del interesado	Rol	Expectativas	Estrategia de comunicación	Frecuencia de comunicación	Responsable de la comunicación
Campos, Cristian	Cliente	Calidad del producto y funcionalidad completa	Presentaciones de avance y demostraciones, reunión aclaración de requerimientos.	Semanal y Quincenal	*Product Owner*
Gutierrez Espinoza, Andres	Desarrollador	Claridad en los requerimientos y soporte *Product Owner*	Reuniones diarias (martes y jueves) y *sprint backlog* de *Jira*	Diaria	*Product Owner*
Fuentes Condega, Jesus	Desarrollador	Claridad en los requerimientos y soporte *Product Owner*	Reuniones diarias (martes y jueves) y *sprint backlog* de *Jira*	Diaria	*Product Owner*

Nombre del interesado	Rol	Expectativas	Estrategia de comunicación	Frecuencia de comunicación	Responsable de la comunicación
Monge, Allen	Calidad de *software*	Claridad en los requerimientos y soporte *Product Owner*	Reuniones diarias (martes y jueves) y *sprint backlog* de *Jira*	Diaria	*Product Owner*

5.2.6.2 Ceremonias regulares

En el contexto de metodologías ágiles, las ceremonias regulares constituyen prácticas estructuradas que permiten mantener la alineación del equipo, fomentar la colaboración continua y asegurar la transparencia en el avance del proyecto. Por tanto, se recomienda calendarizar en la herramienta *teams* las reuniones de la **Cuadro 5.17.** Reuniones del proyecto.

Cuadro 5.17. Reuniones del proyecto

Ceremonia	Objetivo	Frecuencia	Participantes	Herramienta de apoyo
Daily stand-up	Compartir avances, identificar bloqueos y coordinar tareas diarias.	Diaria-máximo 20minutos	Equipo de desarrollo, Product Owner	Microsoft Teams / Jira
Sprint planning	Definir el alcance del *sprint* y estimar las tareas a realizar.	Quincenal -1hora	Equipo de desarrollo, Product Owner	Teams / Jira
Sprint Review	Presentar entregables del *sprint* y recibir retroalimentación de interesados.	Quincenal-1hora	Equipo, Product Owner, *Stakeholders*	Teams
Sprint retrospective	Reflexionar sobre el proceso y definir acciones de mejora.	Quincenal-1hora	Equipo de desarrollo/ Product Owner	Miro / Teams
Backlog refinamiento	Revisar y priorizar elementos del backlog para futuros *sprints*.	Semanal-3horas	Product Owner, Equipo de desarrollo/ Clientes	Jira

5.2.6.3 Herramientas de comunicaciones

En el marco de la solución propuesta, se recomienda la integración de herramientas digitales que permitan la comunicación sincrónica (en tiempo real) y asincrónica (diferida). A continuación, se amplía en detalle las herramientas ya mencionadas en la **sección 5.3.1.2** Definición de herramientas y métodos para el proyecto mediante la **Cuadro 5.18.** Herramientas de comunicación.

Cuadro 5.18. Herramientas de comunicación

Herramienta	Tipo de comunicación	Uso principal	Beneficio clave
Microsoft Teams	Sincrónica / Asincrónica	Reuniones (ceremonias: *dailys, planning, review,* retrospectiva), chats, llamadas, canal temático *Downtime*	Comunicación centralizada y accesible
Outlook / Email	Asincrónica	Acuerdos formales luego de llamadas de *teams*. Seguimientos formales. Notificaciones al cliente.	Registro formal y trazabilidad
Jira	Asincrónica	Backlog del proyecto.	Transparencia en el avance del proyecto
Miro	Sincrónica / Asincrónica	Ceremonia de retrospectiva.	Fomento de la creatividad y participación.

5.2.7 Gestión de riesgos

Para el presente proyecto, se propone una gestión integral de riesgos que abarque todas las fases del ciclo de vida: desde la planificación inicial, pasando por su ejecución, hasta el cierre de este. Esta propuesta surge como respuesta a las lecciones aprendidas en proyectos anteriores, donde se ha identificado que el departamento de Tec Ops presenta debilidades en la aplicación de buenas prácticas relacionadas con la gestión de riesgos.

Con base en esta observación, se establece un enfoque proactivo que busca anticipar, evaluar y mitigar los riesgos potenciales desde el inicio del proyecto. A continuación, se detalla el enfoque propuesto para el manejo de riesgos durante cada una de las etapas del ciclo de vida del proyecto:

- **Inicio del proyecto:** Identificar riesgos iniciales.
- **Planificación del sprint:** Evaluar posible aparición de riesgos en el *sprint*.
- **Ejecución del sprint:** Resolución de riesgos o impedimentos mencionados en las reuniones (*daily o planning*) del equipo durante el *sprint*.
- **Cierre y lanzamiento del sprint:** Revisión del estado de los riesgos.

5.2.7.1 Identificación y manejo de riesgos por sprint

A partir del enfoque de gestión de riesgos previamente establecido, se propone a continuación la integración de ceremonias específicas dentro de cada sprint, en las cuales se debe prestar especial atención a la identificación temprana y al manejo oportuno de posibles riesgos:

- ***Sprint planning:*** Se van a identificar riesgos relacionados al alcance y la capacidad del equipo.
- ***Daily scrum:*** El *Product Owner* debe estar atento a identificar posibles impedimentos o bloqueos que puedan convertirse en riesgos.
- ***Sprint review*:** El *Product Owner* identifica riesgos asociados a la aceptación de algún requerimiento, posibles cambios que deben ser gestionados según sección del documento **5.3.1.8 Gestión de cambios.**
- **Ceremonia de *retrospective*:** El *Product Owner* identifica en la sesión riesgos relacionados con el proceso y el equipo.

5.2.7.2 Registro y seguimiento

La correcta identificación de riesgos es solo el primer paso; su registro, seguimiento y control son esenciales para garantizar una gestión efectiva y proactiva.

En el **Cuadro 5.19.** Escala de probabilidad e impacto, el

Cuadro 5.20. Mapas de calor y **Cuadro 5.21.** *Risk breakdown structure*, son los elementos que constituyen la base metodológica para la identificación, clasificación y evaluación de riesgos, especialmente aquellos relacionados con la calidad del software y la disponibilidad del servicio.

Cuadro 5.19. Escala de probabilidad e impacto

ESCALA		PROBABILIDAD	CRONOGRAMA	COSTO	CALIDAD
5	Muy Alto	>70%	> 6 semanas	-	Impacto muy significativo sobre el alcance del proyecto, que podría ocasionar que no se cumplan los objetivos.
4	Alto	51-70%	5-6 semanas	-	Impacto significativo sobre el alcance del proyecto. Problemas continuos, afectación importante que impacta cumplimiento de objetivos.
3	Mediano	31-50%	3-4 semanas	-	Impacto medio en el alcance, y el cumplimiento de los objetivos se ve comprometido.
2	Bajo	11-30%	1-2 semanas	-	Impacto menor sobre el alcance y el cumplimiento de los objetivos se ve poco comprometido.
1	Muy bajo	1-10%	< 1 semana	-	Impacto mínimo casi nulo en el alcance, y el cumplimiento de los objetivos se ve mínimamente comprometido.
0	Nulo	<1%	Sin cambio	-	Ningún cambio en el alcance del proyecto.

Cuadro 5.20. Mapas de calor

Simbología	
	Evitar
	Mitigar
	Aceptar activamente
	Aceptar pasivamente

Categorías
Técnicos
Externo
Organizacionales
Proyecto

			Impacto en Tiempo					
			Muy Alto	Alto	Mediano	Bajo	Muy bajo	Nulo
			> 6 semanas	5-6 semanas	3-4 semanas	1-2 semanas	< 1 semana	Sin cambio
Probabilidad	Muy Alto	>70%	Evitar	Evitar	Mitigar	Mitigar	Aceptar activam.	Aceptar pasivam.
	Alto	51-70%	Evitar	Evitar	Mitigar	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.
	Mediano	31-50%	Evitar	Mitigar	Aceptar activam.	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.
	Bajo	11-30%	Mitigar	Aceptar activam.	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.
	Muy bajo	1-10%	Aceptar activam.	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.
	Nulo	<1%	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.

			Impacto en calidad					
			Muy Alto	Alto	Mediano	Bajo	Muy bajo	Nulo
			Impacto muy significativo sobre el alcance del proyecto.	Impacto significativo sobre el alcance del proyecto.	Impacto medio en el alcance del proyecto.	Impacto menor sobre el alcance.	Impacto mínimo casi nulo en el alcance.	Ningún cambio en el proyecto.
Probabilidad	Muy Alto	>70%	Evitar	Evitar	Evitar	Evitar o mitigar	Aceptar pasivam.	Aceptar pasivam.
	Alto	51-70%	Evitar	Evitar	Evitar	Mitigar	Aceptar pasivam.	Aceptar pasivam.
	Mediano	31-50%	Evitar	Evitar	Mitigar	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.
	Bajo	11-30%	Mitigar	Mitigar	Mitigar	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.
	Muy bajo	1-10%	Aceptar activam.	Aceptar activam.	Aceptar activam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.
	Nulo	<1%	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.	Aceptar pasivam.

Cuadro 5.21. *Risk breakdown structure*

Nivel 0 de RBS	Nivel 1 de RBS	Nivel 2 de RBS
Todas las fuentes de riesgo del proyecto	1. Riesgo Técnicos	1.1 Fallos de *software*, problemas de integración.
	2. Riesgo Externo	2.1 Clima, Equipos (*Hardware* de la computadora).
	3. Riesgo Organizacional	3.1 Rotación de personal, conflictos internos.
	4. Riesgo Proyecto	4.1 Cronograma, alcance.

La escala de probabilidad e impacto permite asignar valores cualitativos y cuantitativos a cada riesgo, facilitando su priorización. El mapa de calor visualiza el nivel de riesgo según la combinación de probabilidad e impacto, mientras que el RBS organiza los riesgos por categorías, como técnicos, organizacionales, externos y de proyecto.

Con base en esta estructura, se presenta a continuación el **Cuadro 5.22.** Control y seguimiento de riesgos, el cual recoge los riesgos más relevantes mapeados en el inicio del proyecto y continuamente en cada sprint que son asociados a la interrupción del servicio, su evaluación según la escala definida, y las acciones propuestas para su mitigación y seguimiento.

Cuadro 5.22. Control y seguimiento de riesgos

Código	Riesgo	Causa	Consecuencia	Impacto Estimado tiempo (días)	*Plan de Acción	*Responsable
1	El tiempo de los recursos en el proyecto.	Asignación simultánea de los recursos a múltiples proyectos o tareas operativas.	Retraso en la entrega de las tareas en los *sprints*.	3		
2	La adopción de la nueva herramienta en el piso de producción	Resistencia al cambio por parte del personal operativo.	Retrasos en la implementación operativa y disminución de la eficiencia productiva.	10		
3	Aprobación del proyecto por el equipo de "*Architect Engagement Evaluation"*	No se explican los objetivos del proyecto de forma clara.	Retraso en la aprobación del proyecto del proyecto.	5		

Código	Riesgo	Causa	Consecuencia	Impacto Estimado tiempo (días)	*Plan de Acción	*Responsable
4	Fallos de *software* en producción	Errores no detectados durante pruebas unitarias	Interrupción del servicio, insatisfacción del usuario	3		
5	Problemas de integración entre módulos	Falta de alineación en los estándares de desarrollo	Retrasos en la entrega, aumento de costos por retrabajo	3		
6	Uso de tecnologías obsoletas	Dependencia de sistemas heredados	Dificultad para mantener y escalar el sistema	3		
7	Rotación de personal clave	Falta de motivación o mejores ofertas externas	Pérdida de conocimiento, disminución de la productividad	10		
8	Conflictos internos en el equipo	Diferencias en prioridades o estilos de trabajo	Ambiente laboral tenso, retrasos en tareas colaborativas	10		

Código	Riesgo	Causa	Consecuencia	Impacto Estimado tiempo (días)	*Plan de Acción	*Responsable
9	Falta de compromiso del *Product Owner*	Sobrecarga de responsabilidades o falta de disponibilidad	Decisiones tardías, bloqueos en el desarrollo	10		
10	Comunicación deficiente entre áreas	Falta de canales formales o reuniones de coordinación	Malentendidos, duplicación de esfuerzos	8	Actualizar el cronograma, sumando estos 8 días de impacto que no afectan aun la fecha de entrega. Convocar reunión extraordinaria de alineamiento y documentar acuerdos alcanzados.	*Product Owner*
11	Fallos en el *hardware* de la computadora	Accidentes con las computadoras	Pérdida de datos, interrupción de actividades críticas	5		

Código	Riesgo	Causa	Consecuencia	Impacto Estimado tiempo (días)	*Plan de Acción	*Responsable
12	Condiciones climáticas adversas	Tormentas, inundaciones, etc.	Imposibilidad de acceso físico a la oficina, retrasos en el proyecto	3		
13	Desviación del cronograma	Subestimación del esfuerzo requerido	Entrega tardía, no se completaron tareas del sprint	5	Actualizar el cronograma, sumando estos 5 días de impacto que no afectan aun la fecha de entrega.	*Product Owner*
14	Cambios frecuentes en el alcance	Requerimientos poco definidos o cambios por parte del cliente	Incremento de costos, disminución de la calidad del producto	10	Negociar con el cliente los requerimientos nuevos que se pueden ejecutar revisando con los tres *sprints* que se tienen de Mejoras.	*Product Owner*

Código	Riesgo	Causa	Consecuencia	Impacto Estimado tiempo (días)	*Plan de Acción	*Responsable
15	Falta de definición de criterios de aceptación	Documentación incompleta o ambigua	Rechazo de entregables, retrabajo	3	Revisar con el cliente los criterios de aceptación para próximos sprint.	*Product Owner*

**Nota*: Los riesgos con las columnas de “Plan de Acción” y “Responsable” vacíos es porque fueron aceptados pasivamente entonces no es necesaria la información.

5.2.8 Gestión de cambios

La agilidad tiene la particularidad de que todo cambio es "abrazado", se permite la mejora continua dentro del proceso de desarrollo, pero la experiencia en proyectos anteriores ha determinado que los clientes en Boston Scientific tienen la tendencia de sugerir más cambios de los esperados por el equipo y esto en muchas ocasiones ha tenido como resultado una entrega tardía con respecto al calendario inicial. La particularidad de ***Downtime*** es que de acuerdo con el inicio del proyecto en junio 16 y luego de la estimación de las historias de usuario se tiene un tiempo adecuado para desarrollar algunos requerimientos de la épica "Mejora". La recomendación será desarrollar algunos de acuerdo con el alcance previamente definido.

Actividades relacionadas con la gestión de cambios dentro del ciclo de vida del proyecto:

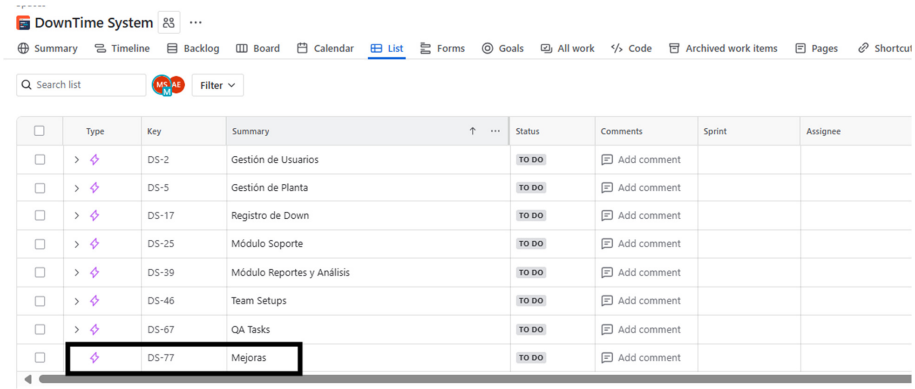
- **Seguimiento y control:** Adaptación e integración al backlog.

5.2.8.1 Adaptación e integración al backlog

Con el objetivo de asegurar una adecuada gestión de las mejoras identificadas durante el proceso de revisión del *sprint* (*sprint review*), se establece el siguiente mecanismo de adaptación e integración al *backlog* utilizando la herramienta *Jira*:

- **Organización del *backlog*:** El *backlog* del producto se encuentra estructurado por épicas, lo que permite una mejor categorización y trazabilidad de los elementos. Se recomienda la creación de una épica específica denominada "Mejoras" la cual se puede visualizar en la **Figura 5.7.** Épica "Mejoras" , en la cual se incluirán todos los ítems que surjan como retroalimentación durante las ceremonias de revisión del *sprint*. Esta épica funcionará como un repositorio centralizado para las oportunidades de mejora continua del producto.

Figura 5.7. Épica “Mejoras”



- **Evaluación e inclusión en el *sprint*:** Durante la planificación del *sprint* (*sprint planning*), el *Product Owner* y el equipo de desarrollo deberán revisar los ítems contenidos en la épica “Mejoras” y evaluar su viabilidad para ser incluidos en el *sprint* siguiente. Esta evaluación debe considerar:
    - Prioridad del ítem según valor de negocio y urgencia.
    - Esfuerzo estimado y capacidad del equipo.
    - Impacto en el cronograma y dependencias con otros ítems.

  En caso de que se apruebe la inclusión de mejoras en el sprint, el *Product Owner* deberá actualizar el cronograma del proyecto conforme a lo establecido en la sección **5.3.1.3.3** Control del Cronograma, asegurando que los cambios sean comunicados oportunamente a los *stakeholders*.
- **Seguimiento y trazabilidad:** Cada mejora incluida en el *backlog* debe contar con una descripción clara, criterios de aceptación definidos y una etiqueta que permita su fácil identificación dentro de la épica “Mejoras”. Se recomienda utilizar etiquetas como retroalimentación, mejora continua, o *post-sprint review* para facilitar su seguimiento.
- **Beneficios de la Integración:** Este enfoque permite:
    - Fomentar la mejora continua del producto.

- Aumentar la transparencia en la gestión de retroalimentación.
- Facilitar la toma de decisiones durante la planificación.
- Mantener la alineación entre el equipo de desarrollo y el *Product Owner.*

5.2.9 Cierre del proyecto

El proyecto se considerará oficialmente concluido cuando se hayan cumplido los criterios de aceptación del MVP definidos en cada historia de usuario en la herramienta de *Jira,* estos son revisando dentro de las pruebas de QA.

Actividades del cierre del proyecto:

5.2.9.1 Lanzamiento en el ambiente de producción

Primero el equipo de desarrollo recibe por parte del cliente la información requerida para la base de datos de producción, el equipo la orden y actualiza la base de datos, luego debe ejecutar el lanzamiento de la aplicación en el servidor de producción. Entonces así la aplicación ya estará disponible para el cliente y resto de la población de la sede de Heredia de Boston Scientific que debe utilizar la aplicación.

5.2.9.2 Retrospectiva del proyecto

La sesión de retrospectiva del proyecto resulta un componente esencial en la fase de cierre del proyecto, ya que permite al equipo reflexionar de manera estructurada sobre la ejecución completa del mismo. Su propósito principal es identificar y documentar los aciertos, los desafíos enfrentados y las oportunidades de mejora, con el fin de fortalecer la gestión y desempeño en futuros proyectos.

Se recomienda estructurar la sesión de la siguiente manera:

- Revisión de objetivos y entregables
  - Comparar lo que se planeó vs lo que se entregó.
  - Verificar si se cumplieron los criterios de éxito del proyecto.
  - Identificar desviaciones y sus causas.
- Recopilación de opiniones del equipo
  - ¿Qué funcionó bien?
  - ¿Qué no funcionó?

- ¿Qué se podría mejorar?

Estas preguntas se responden mediante la herramienta Miro.

- Lecciones aprendidas
  - Que haríamos diferente la próxima vez.
  - Qué proceso o herramienta mejoraríamos.
  - Como evitar errores identificados.
- Retroalimentación del equipo
  - Como se sintió durante el proyecto.
  - Que aprendió.
  - ¿Sugerencias?
- Reconocimiento de logros del equipo.

Luego de la sesión se debe completar la plantilla del cierre del proyecto que sería el **Cuadro 5.23.** Plantilla informe cierre *Downtime*.

Cuadro 5.23. Plantilla informe cierre *Downtime*

Problema:		
Resumen del proyecto:		
Roles y Responsabilidades		
Alcance	**Planeado**	**Real**
Entregables:	**Planeado**	**Real**
Informe de Calidad		
Informe de riesgos	**Identificados**	**Materializados**

Lecciones aprendidas		
Representante del sistema para auditorias:		
Aprobación:	**Nombre y Rol**	**Firma**

5.2.9.3 Firma del informe cierre del proyecto

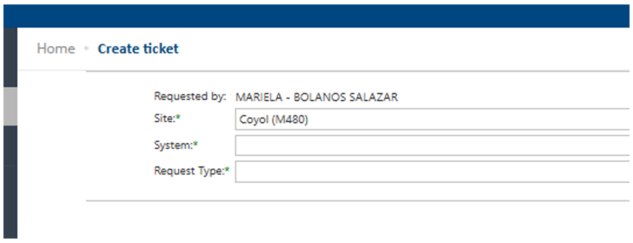
Al completar la plantilla se debe enviar por la herramienta de *Adobe* para recopilar las firmas debidas.

5.2.9.4 Soporte del sistema

Todas las aplicaciones locales tienen soporte y deben reportar cualquier problema creado un reporte de incidente mediante la plataforma de Tec Ops **Figura 5.8.** Tec Ops creación de incidente**,** seleccionar a continuación:

- Site: Heredia
- Sistema: “Downtime”
- Tipo de soporte: “Soporte”

Figura 5.8. Tec Ops creación de incidente



5.3 Forma de trabajar del equipo

A partir del análisis realizado en el capítulo 4, se concluye que el enfoque más adecuado para este proyecto es trabajar bajo un modelo predominante ágil con algunos componentes de la

gestión predictiva de proyectos, específicamente utilizando el marco de trabajo *Scrum*, debido a que el equipo de proyecto ya ha trabajado con varios de los elementos de Scrum.

Se considera que para la ejecución del proyecto es necesario contar con 3 roles que tienen responsabilidades definidas los cuales se detallan a continuación:

- ***Product Owner/ Scrum Master:*** Dado el personal disponible en el equipo de software, se propone que una misma persona asuma ambos roles: *Product Owner y Scrum Master*. Esta decisión busca optimizar recursos sin comprometer la eficiencia del proyecto. Responsabilidades clave:
    - Actuar como punto de contacto entre el patrocinador y el equipo de desarrollo.
    - Gestionar y priorizar el backlog del producto, asegurando que las tareas estén alineadas con los objetivos del negocio.
    - Coordinar las ceremonias *Scrum*, incluyendo:
        - *Daily Standups*
        - *Sprint Planning*
        - *Sprint Review*
        - *Retrospective*
- **Equipo de desarrollo:** Conformado por profesionales técnicos con las habilidades necesarias para:
    - Ejecutar la programación del sistema.
    - Realizar pruebas de *Quality Assurance* (QA).
    - Validar que se cumplan los criterios de aceptación definidos para cada entregable.

  Este equipo debe trabajar de forma colaborativa, manteniendo una comunicación constante y transparente, y adaptándose a los cambios que surjan durante el desarrollo.
- **Patrocinador/ Clientes:** Nick representa el patrocinador y los clientes son representados por los departamentos interesados: Materiales, Equipos e Industrial. Su rol es fundamental para el éxito del proyecto, ya que deben:
    - Proporcionar retroalimentación continua sobre los entregables.

- Validar que cada entrega cumpla con los criterios de aceptación definidos.
- Participar activamente en las revisiones de *sprint* y en la definición de prioridades.
- Por parte del patrocinador se espera la ayuda necesaria si los clientes no muestran el interés esperado.

5.3.1 Dinámica de trabajo

El equipo trabajará en *sprints* cortos (cada 2 semanas), con entregables incrementales que permitan validar avances de forma continua. Se fomentará la colaboración interdisciplinaria, la adaptabilidad ante cambios y la transparencia en la comunicación.

Además, se incorporarán herramientas y métricas específicas que permitirán controlar de manera efectiva el avance del proyecto y evaluar el desempeño del equipo. Estas herramientas facilitarán el seguimiento de tareas, la visualización del progreso en tiempo real y la identificación de posibles desviaciones respecto al cronograma establecido. Las métricas, por su parte, ofrecerán indicadores clave sobre la productividad, la velocidad de entrega, la calidad de los entregables y la eficiencia en la resolución de impedimentos, lo que contribuirá a una toma de decisiones más informada y oportuna durante cada sprint. Para detalle ver **sección 5.3.1.2** Definición de herramientas y métricas para el proyecto.

Documentación del proyecto: agregar algo que describa como se llevará esto

Capítulo 6 Conclusiones y Recomendaciones

6.1 Conclusiones

- La caracterización detallada del proyecto "*Downtime System*", a través del levantamiento de requerimientos funcionales y no funcionales, permitió delimitar claramente el alcance y las funcionalidades esperadas. Esto facilitó la alineación de expectativas entre los interesados y el equipo de desarrollo, sentando una base sólida para la planificación y ejecución del proyecto.
- El análisis y selección de buenas prácticas, sustentado en la revisión bibliográfica y la experiencia del equipo, permitió adaptar metodologías ágiles (principalmente Scrum) al contexto del proyecto. Esto favoreció la flexibilidad ante cambios, la colaboración continua y la entrega incremental de valor, incrementando la probabilidad de éxito en la implementación del sistema.
- La elaboración de un plan de gestión que incluyó planes subsidiarios (alcance, cronograma, costos, calidad, riesgos, comunicaciones, entre otros) y el uso de herramientas colaborativas como Jira y Miro, contribuyó a una gestión eficiente de tareas, riesgos y calidad. Este enfoque permitió anticipar y mitigar posibles desviaciones, asegurando la entrega oportuna del sistema.
- El establecimiento de métricas clave y mecanismos de seguimiento, como reuniones periódicas y tableros de control, facilitó la toma de decisiones correctivas a tiempo durante la ejecución del proyecto. Además, la documentación sistemática de lecciones aprendidas y la evaluación constante del desempeño promovieron una cultura de mejora continua en el departamento Tec Ops.

6.2 Recomendaciones

- Para futuros proyectos en el departamento Tec Ops, se recomienda mantener una metodología ágil o mixta dado el entorno de la organización (quieren ser agiles, pero aún no están preparados) y fortalecer la capacitación en gestión de proyectos

para todos los miembros del equipo, especialmente en roles clave como *Product Owner* y *Scrum Master.*

- Es fundamental dentro del departamento Tec Ops documentar de manera sistemática las lecciones aprendidas y crear repositorios accesibles para facilitar la transferencia de conocimiento y evitar errores recurrentes.
- Se sugiere con la ayuda del sponsor Nick O'Shea ampliar la integración del sistema *Downtime* a otros departamentos y sites, considerando las particularidades de cada área y promoviendo la estandarización de procesos.
- El Product Owner y/o Scrum Master deberá continuar con la evaluación y ajuste de métricas de desempeño, calidad y costos, asegurando la mejora continua en el proceso de entregables. Este enfoque garantizará la alineación con los objetivos estratégicos de la organización y la satisfacción de los usuarios finales en cuanto a tiempos, adaptabilidad y soporte de la herramienta.

Capítulo 7 Referencias Bibliográficas

Bermejo, M. (2011). El Kanban. Barcelona, España: UOC, 8.

Conecta Software. (2025). *10 Mejores Herramientas de Gestión de Proyectos.* Conecta Magazine. https://www.conectasoftware.com/magazine/herramientas-de-gestion-de-proyectos/

Díaz Alonso, L. M., & Delgado Olivera, L. de la C. (2021). *Modelos de desarrollo de software*. Revista Cubana de Ciencias Informáticas, 15(1), 38-39. http://scielo.sld.cu/pdf/rcci/v15n1/2227-1899-rcci-15-01-37.pdf

Foces Vivancos, M. Á., Pérez Carvajal, M. L., Crespo, J. A., Marambio, V. L., & Hernández H., A. A. (2021). *La agilidad en América latina*. MIT Technology Review en español.

Ibrahim, R., Hamou-Lhadj, A., & Lakhotia, A. (2021). *Automating software maintenance with machine learning techniques.* IEEE Transactions on Software Engineering.

ISO. (2017). *ISO 10006: Directrices para la gestión de la calidad en proyectos*. Plataforma de navegación en línea (OBP). https://www.iso.org/obp/ui/es/#iso:std:iso:10006:ed-3:v1:es

ISO. (2020). *ISO 21502: Project, programme and portfolio management.* Plataforma de navegación en línea (OBP). https://www.iso.org/obp/ui/es/#iso:std:iso:21502:ed-1:v1:en

Meléndez, J. R., & El Salous, A. (2021). *Factores críticos de éxito y su impacto en la gestión de proyectos empresariales*. Revista de Ciencias Sociales. https://dialnet.unirioja.es/servlet/articulo?codigo=8229889

Lanús, M. (2016). *Gerencia de proyectos*. https://d1wqtxts1xzle7.cloudfront.net/43470390/v10n2a03-libre.pdf?1457382414=&response-content-disposition=inline%3B+filename%3DGERENCIA_DE_PROYECTOS.pdf&Expires=17173

54015693&Signature=Y3jAuBbQoaG40J4rBmOmMKJ-tpQTYKxf6UaZk6UQIYiu02S8Dk1rKpLm6tU6v94aPRi0qH8y0erawpGnSZmPm-B29WaZgfWKUia0lOLUDFTm0gADae6VFmyT-b0IKQc8iU0vTJNrWLr8HB-ju-NXZV6CXYKWuq3B9pOGdPm-gzLIHBtrEHmXfRfBV0XSnjK4HXeShvYkgx4l4fWd2-0tzRjR0UL9B~ciHlyBEN4ucxw8mvb4yMtg1CGeK1suTSdT5p6s83kNSc9E4GDev6XnDW49AmtNe3lC1D5okR~ZejI6wE-kkkX3ebhUsKoYXHL5cQl4UsfEt2DejWYlI2ZZiQ__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

Lohr, S. L. (2022). *Sampling: Design and Analysis (3rd ed.).* Chapman and Hall/CRC.

Project Manager Institute. (2017). *La Guía de los fundamentos para la dirección de proyectos* (Guía del *PMBOK* 6ta edición) (pp.23-24). Project Management Institute, Inc.

Project Manager Institute. (2021). *Guía de los fundamentos para la dirección de proyectos* (Guía del *PMBOK* 7ma edición) (pp.4). Project Management Institute, Inc.

Project Manager Institute. (2021). *Guía de los fundamentos para la dirección de proyectos* (Guía del *PMBOK* 7ma edición) (pp.33). Project Management Institute, Inc.

Project Manager Institute. (2021). *Guía de los fundamentos para la dirección de proyectos* (Guía del *PMBOK* 7ma edición) (pp.35). Project Management Institute, Inc.

Project Manager Institute. (2021). *Guía de los fundamentos para la dirección de proyectos* (Guía del *PMBOK* 7ma edición) (pp.42). Project Management Institute, Inc.

Project Management Institute, Inc. (2018). *Guía práctica ágil* (pp.8-9). Project Management Institute, Inc.

Pérez, A. (2021). *Tipos de proyectos y sus principales características*. obsbusiness.school. https://www.obsbusiness.school/blog/tipos-de-proyectos-y-sus-principales-caracteristica

Sánchez Huarcaya, A. (2020). *LOS MÉTODOS DE INVESTIGACIÓN PARA LA ELABORACIÓN DE LAS TESIS DE MAESTRÍA EN EDUCACIÓN*. PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ. https://repositorio.pucp.edu.pe/items/9509133a-a5d1-4594-baf4-c61775ea4b19

SCRUMstudy. (2022). *Guía de los fundamentos de Scrum* (Guía del SBOK Cuarta edición) (pp. 2). SCRUMstudy™. www.scrumstudy.com

Scrum.org, Daniel Vacanti, & Yuval Yeret. (2021). *La guía Kanban para Scrum Teams*. Scrum.org.

Veloz, E. Veloz, V. (2024). *Tendencias y prácticas actuales en Ingeniería del Software*, G-ner@ndo, V°5 (N°2,). 2509 – 2515.

Capítulo 8 Apéndices

Apéndice A: Revisión bibliográfica

Revisión bibliográfica	
Tema:	Nombre del documento:
Autor (es):	Ano de publicación:
Edición:	Editorial:
Resumen:	Comentarios:
Análisis / información recolectada:	

Apéndice B: Entrevista semiestructurada

Fecha:	
Nombre de la persona entrevistada	
Tipo de Participante	Departamentos interesados
Nombre del entrevistador	
Método de aplicación	☐Individual ☐ Grupal
Técnica de Aplicación	☐Presencial ☐ Virtual

Guía de preguntas para el participante

Departamentos interesados
1- ¿Cuáles son las principales limitaciones del proceso actual (VTP (Supplier Quality), PTS (Equipos), Line Disruption (Industrial))?
2- ¿Qué indicadores le muestran que existe un problema con el proceso actual?
3- ¿Qué funciones espera que el sistema realice?
4- ¿Qué tareas desea que se automaticen con este sistema?
5- ¿Qué tipo de reportes o métricas necesita que el sistema genere?
6- ¿Qué funcionalidades considera indispensables desde el primer lanzamiento del sistema?

7- ¿Qué función del sistema debería resolver el mayor dolor o problema actual?
8- ¿Qué funcionalidades impactarían más positivamente en su trabajo diario?
9- ¿Quiénes serían los administradores del sistema?
10- ¿El dueño del sistema en caso de auditoria seria?

Apéndice C: Entrevista abierta

Fecha:	
Nombre de la persona entrevistada	
Tipo de Participante	Desarrolladores con roles de Project Managers en otros proyectos
Nombre del entrevistador	
Método de aplicación	☐Individual ☐ Grupal
Técnica de Aplicación	☐Presencial ☐ Virtual

Guía de preguntas para el participante

Desarrolladores con roles de Project Managers en otros proyectos
1- ¿Podrías contarme un poco de tu experiencia en proyectos donde has sido figura de Project Manager?
2- ¿Qué herramientas has utilizado para la dirección de proyectos?
3- ¿Qué dificultades recuerdas se han materializados en medio del proyecto y como las abordaste?
4- ¿Qué acciones consideras efectivas en el contexto actual?
5- ¿Qué metodologías has aplicado y por qué?
6- ¿Has identificado prácticas que generan problemas o limitaciones?
7- ¿Qué sugerencias tienes para mejorar las prácticas actuales?

Apéndice D: Talleres de focus group

Fecha	
Nombre de las personas asistentes	
Tipo de Participantes	Departamentos interesados y Equipo técnico asignado al proyecto
Nombre del host de la sesión	
Técnica de aplicación	☐Presencial ☐ Virtual
Minuta de la sesión:	

Capítulo 9 Anexos

Anexo 1: Elección ciclo de vida

Filtro de idoneidad

Cultura **Aceptación del enfoque**	**Calificar**
1- ¿Existe un patrocinador sénior que entienda y apoye el uso de un enfoque ágil para este proyecto? 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	1
Confianza en el equipo	
2- Tomando en cuenta los patrocinadores y los representantes del negocio que trabajarán con el equipo. ¿Tienen estos interesados la confianza en que el equipo puede transformar su visión y necesidades en un producto o servicio exitoso, con apoyo y retroalimentación continuos en ambas direcciones? 1 corresponde: SI, 5 corresponde: PROBABLEMENTE, 10 corresponde: POCO PROBABLE	1
Poderes del equipo para la toma de decisiones	
3- ¿Se le dará autonomía al equipo para tomar sus propias decisiones locales sobre cómo emprender el trabajo? 1 corresponde: SI, 5 corresponde: PROBABLEMENTE, 10 corresponde: POCO PROBABLE	3
Equipo **Tamaño del equipo**	
4- ¿Cuál es el tamaño del equipo principal? Usar esta escala: 1-9 = 1, 10-20 = 2, 21-30 = 3, 31-45 = 4, 46-60 = 5, 61-80 = 6, 81-110 = 7, 111-150 = 8, 151 – 200 = 9, 201+ = 10	1
Niveles de experiencia	

5- Considerar los niveles de experiencia y habilidades de los roles del equipo principal. Aunque es normal tener una mezcla de personas experimentadas e inexpertas en los roles, para que los proyectos ágiles funcionen sin problemas es más fácil cuando cada rol tiene al menos un miembro experimentado 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	2
Acceso al cliente/ Negocio	
6- ¿Tendrá el equipo acceso diario a por lo menos un representante del negocio/del cliente con el fin de hacer preguntas y obtener retroalimentación? 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	2
Proyecto **Probabilidad de cambio**	
7- ¿Qué porcentaje de requisitos podrían cambiar o ser descubiertos mensualmente? 1 corresponde: 50%, 5 corresponde: 25%, 10 corresponde: 5%	3
Criticidad del producto o servicio	
8- Para ayudar a determinar los niveles probables de rigor adicional para verificación y documentación que puedan requerirse, evaluar la criticidad del producto o servicio que se está construyendo. Utilizando una evaluación que considere pérdidas debida al posible impacto de los defectos, determinar que podría ocasionar una falla. 1 corresponde: Tiempo, Fondos Discrecionales, 5 corresponde: Fondos Esenciales, Vida única, 10 corresponde: Muchas vidas	2
Entrega incremental	
9- ¿Se puede construir y evaluar el producto o servicio en porciones? Además, ¿estarán disponibles los representantes de la empresa o del cliente para proporcionar retroalimentación oportuna sobre los incrementos entregados?	1

1 corresponde: SI, 5 corresponde: QUIZÁS / A VECES, 10 corresponde: POCO PROBABLE	

Anexo 2: Resultado Entrevistas

Entrevista Semiestructurada

Fecha: 5- de julio 2025	
Nombre de la persona entrevistada	Dennis Villalobos, Cristian Campos, Cesar Carrillo, Yecson Vargas, Michelle Tenorio.
Tipo de Participante	Departamentos interesados
Nombre del entrevistador	**Mariela Bolaños**
Método de aplicación	☐Individual ☒ Grupal
Técnica de Aplicación	☐Presencial ☒ Virtual

Departamentos interesados
1- ¿Cuáles son las principales limitaciones del proceso actual (VTP (Supplier Quality), PTS (Equipos), Line Disruption (Industrial))? • No tener conexión entre los sistemas. • PTS los reportes no funcionan correctamente o del todo no cargan la información. • En Line disruption no se obtiene la información veraz ya que los chicos a veces no registran el down acá. • Por parte de los chicos del piso olvidan documentar los down porque se confunden entre sistemas. • Las herramientas de VTP y PTS no se pueden cambiar o mejorar porque son softwares que no tenemos soporte local o corporativo.
2- ¿Qué indicadores le muestran que existe un problema con el proceso actual? En las métricas de los tres sistemas existen diferencias en los down reportados.
3- ¿Qué funciones espera que el sistema realice? Que se puedan crear downs centralizados en un solo lugar. Que pueda editar la hora del cierre del ticket. Que envíe notificaciones en los diferentes estatus del ticket. Que permita la reasignación del ticket a otro equipo si fuera necesario. Que pueda dar seguimiento a los tickets de materiales agregando el plan de acción ejecutado.

Debe tener los actuales reportes, pero además queremos nuevos reportes que solemos crear nosotros de forma manual. Que pueda customizar los campos en la gestión del ticket.
4- ¿Qué tareas desea que se automaticen con este sistema? Él envió de notificaciones. El seguimiento en el caso de materiales “Plan de acción”. La generación de los nuevos reportes.
5- ¿Qué tipo de reportes o métricas necesita que el sistema genere? Los actuales, pero además nuevos: Reportes tickets del día en Equipos. Reporte por “Vendor” y horas detenidas de la unidad de negocio. Reporte de unidades perdidas, dinero perdido y el motivo del down.
6- ¿Qué funcionalidades considera indispensables desde el primer lanzamiento del sistema? Las que se dijo anteriormente y los reportes.
7- ¿Qué función del sistema debería resolver el mayor dolor o problema actual? La creación del ticket down centralizado.
8- ¿Qué funcionalidades impactarían más positivamente en su trabajo diario? Para materiales los reportes nuevos y el plan de acción. Para equipos el estado de following y reportes nuevos. El nuevo reporte que incluye las unidades perdidas y el dinero.
9- ¿Quiénes serían los administradores del sistema? En términos del negocio seriamos nosotros mismos.
10- ¿El dueño del sistema en caso de auditoria seria? En términos del negocio seriamos nosotros mismos.

Entrevista abierta

Fecha: 7 de Julio	
Nombre de la persona entrevistada	**Jeifer Porras, Daniela Araya, Esteban Chaves**
Tipo de Participante	Desarrolladores con roles de Project Managers en otros proyectos
Nombre del entrevistador	**Mariela Bolaños**
Método de aplicación	☐Individual ☒ Grupal
Técnica de Aplicación	☐Presencial ☒ Virtual

Desarrolladores con roles de Project Managers en otros proyectos
1- ¿Podrías contarme un poco de tu experiencia en proyectos donde has sido figura de Project Manager? Bueno tenemos clientes en este negocio un poco intensos, dado el negocio pasa que quieren las entregas en tiempos que no son factibles o por ejemplo cambian el requerimiento inicial dado cuando estamos en la revisión del sprint y muchas veces cambia mucho lo que ya tenía el equipo de desarrollo. También algunos clientes tienen urgencias en sus departamentos y no acuden a las reuniones de requerimientos.
2- ¿Qué herramientas has utilizado para la dirección de proyectos? Jira que es la licencia que tenemos, nos apoyamos con teams y sus herramientas, correos y Miro a veces.
3- ¿Qué dificultades recuerdas se han materializados en medio del proyecto y como las abordaste? La no asistencia del cliente recurrentemente a algunas reuniones y bueno por dicha nuestros proyectos tienen sponsor y lo que hemos realizado es escalar a tiempo este tema. La entrega tardía de algún requerimiento, pero se ha valorado alinear el siguiente sprint.
4- ¿Qué acciones consideras efectivas en el contexto actual?

Las escalaciones. Traer recursos si fuera necesario.
5- ¿Qué metodologías has aplicado y por qué? Antes hacíamos todo cascada, pero hemos venido evolucionando y ahora lo hacemos hibrido entre agiles y cascada.
6- ¿Has identificado prácticas que generan problemas o limitaciones? A veces la asignación en Jira ya que no se tiene la visualización en el correo como es lo esperado.
7- ¿Qué sugerencias tienes para mejorar las prácticas actuales? Deberíamos compartir las lecciones aprendidas en los Project Manager o Product Owners.

Anexo 2: Talleres focus group

Sesión alineamiento Sprint #1

Fecha	01 junio 2025
Nombre de las personas asistentes	Dennis Villalobos, Cristian Campos, Cesar Carrillo, Yecson Vargas, Michelle Tenorio.
Tipo de Participantes	Departamentos interesados y Equipo técnico asignado al proyecto
Nombre del host de la sesión	Mariela Bolaños
Técnica de aplicación	☐Presencial ☒ Virtual
Minuta de la sesión: **Se acordó entrega de "Registrar paros en general"** **Se explica durante este sprint muchas tareas ejecutadas por los desarrolladores no son visibles, pero con ello estamos adelantando trabajo en base de datos para las futuras tareas, siendo preventivos antes las "Mejoras".**	

Anexo 3: Filtros de idoneidad

Fecha:	
Participantes:	
Forma de aplicación:	☐Virtual ☐Presencial

Comentado [AHG7]: Eliminar en todo el documento lo que indique "Elaboración propia", en APA 7 lo realizado por el autor no debe indicar nada.

Instrucciones de uso:

1- Trabajo colaborativo en el cuestionario: El cuestionario debe completarse en conjunto por el cliente y el equipo técnico. La intención es fomentar una evaluación grupal de cada pregunta, promoviendo el intercambio de perspectivas y asegurando que todas las respuestas reflejen un consenso informado

2- Escala de valoración del 1 al 10: Cada pregunta se debe puntuar en una escala del 1 al 10. Se incluirá una explicación clara de lo que representa cada valor dentro de la escala, para facilitar una interpretación coherente y alineada entre todos los participantes.

3- Análisis e interpretación de resultados: Una vez completado el cuestionario, se interpretarán los resultados de forma grupal. Este análisis permitirá tomar una decisión conjunta sobre el ciclo de vida más adecuado para el proyecto, considerando tanto las necesidades del cliente como las capacidades técnicas del equipo.

Filtro de idoneidad	
Cultura **Aceptación del enfoque**	**Calificar**
1- ¿Existe un patrocinador sénior que entienda y apoye el uso de un enfoque ágil para este proyecto? 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	
Confianza en el equipo	
2- Tomando en cuenta los patrocinadores y los representantes del negocio que trabajarán con el equipo. ¿Tienen estos interesados la confianza en que el equipo puede transformar su visión y necesidades en un producto o servicio exitoso, con apoyo y retroalimentación continuos en ambas direcciones? 1 corresponde: SI, 5 corresponde: PROBABLEMENTE, 10 corresponde: POCO PROBABLE	
Poderes del equipo para la toma de decisiones	

3- ¿Se le dará autonomía al equipo para tomar sus propias decisiones locales sobre cómo emprender el trabajo? 1 corresponde: SI, 5 corresponde: PROBABLEMENTE, 10 corresponde: POCO PROBABLE	
Equipo **Tamaño del equipo**	
4- ¿Cuál es el tamaño del equipo principal? Usar esta escala: 1-9 = 1, 10-20 = 2, 21-30 = 3, 31-45 = 4, 46-60 = 5, 61-80 = 6, 81-110 = 7, 111-150 = 8, 151 – 200 = 9, 201+ = 10	
Niveles de experiencia	
5- Considerar los niveles de experiencia y habilidades de los roles del equipo principal. Aunque es normal tener una mezcla de personas experimentadas e inexpertas en los roles, para que los proyectos ágiles funcionen sin problemas es más fácil cuando cada rol tiene al menos un miembro experimentado 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	
Acceso al cliente/ Negocio	
6- ¿Tendrá el equipo acceso diario a por lo menos un representante del negocio/del cliente con el fin de hacer preguntas y obtener retroalimentación? 1 corresponde: SI, 5 corresponde: PARCIAL, 10 corresponde: NO	
Proyecto **Probabilidad de cambio**	
7- ¿Qué porcentaje de requisitos podrían cambiar o ser descubiertos mensualmente? 1 corresponde: 50%, 5 corresponde: 25%, 10 corresponde: 5%	
Criticidad del producto o servicio	

8- Para ayudar a determinar los niveles probables de rigor adicional para verificación y documentación que puedan requerirse, evaluar la criticidad del producto o servicio que se está construyendo. Utilizando una evaluación que considere pérdidas debida al posible impacto de los defectos, determinar que podría ocasionar una falla. 1 corresponde: Tiempo, Fondos Discrecionales, 5 corresponde: Fondos Esenciales, Vida única, 10 corresponde: Muchas vidas	
Entrega incremental	
9- ¿Se puede construir y evaluar el producto o servicio en porciones? Además, ¿estarán disponibles los representantes de la empresa o del cliente para proporcionar retroalimentación oportuna sobre los incrementos entregados? 1 corresponde: SI, 5 corresponde: QUIZÁS / A VECES, 10 corresponde: POCO PROBABLE	

Anexo 4: Revisión bibliográfica

Revisión bibliográfica	
Tema: Scrum	Nombre del documento: The Scrum Guide (Definitive Guide to Scrum)
Autor (es): Ken Schwaber; Jeff Sutherland	Ano de publicación: 2020
Edición: Nov 2020 (Versión actual)	Editorial: ScrumGuides.org
Resumen: Define oficialmente Scrum, sus responsabilidades, eventos, artefactos y compromisos (Product Goal, Sprint Goal, Definition of Done).	Comentarios: Texto fundacional, menos prescriptivo en 2020; útil para equipos que requieren claridad del marco.

Análisis / información recolectada: Características: marco ligero basado en empirismo y pensamiento lean; sprints de 2–4 semanas. Ventajas: foco en valor, inspección y adaptación frecuente, transparencia. Desventajas: requiere equipos maduros; puede ser difícil de escalar sin prácticas adicionales.

Revisión bibliográfica	
Tema: Kanban (para equipos Scrum)	Kanban Guide for Scrum Teams
Autor (es): Scrum.org (Porter; Vacanti; Yeret, y comunidad)	Ano de publicación: 2021
Edición: Actualización enero 2021	Editorial: Scrum.org
Resumen: Guía para complementar Scrum con prácticas de flujo: visualización, límites de WIP, gestión activa del trabajo en progreso y definición de workflow.	Comentarios: Relevante cuando se busca mejorar el flujo y la predictibilidad sin cambiar Scrum.
Análisis / información recolectada: Características: optimización del flujo, métricas de flujo (lead time, cycle time). Ventajas: mejor transparencia, detección de cuellos de botella, entrega continua. Desventajas: menor énfasis en planificación a largo plazo; puede sobresimplificar dependencias si no se gestiona bien.	

Anexo 5: Seguimiento proyecto Downtime System. Portafoloio Transformación Digital Heredia.

