

INSTITUTO TECNOLÓGICO DE COSTA RICA

ÁREA ACADÉMICA DE GERENCIA DE PROYECTOS

MAESTRÍA EN GERENCIA DE PROYECTOS



Método de gestión de requisitos de proyectos para la Tribu Navi

Proyecto de graduación para optar por el grado académico de

Maestría en Gerencia de Proyectos.

Realizado por:

Mayid Nasralah Calderón

San José, 15 de enero de 2022

DEDICATORIA

Este logro es dedicado a mi esposa Eva, quien ha estado a mi lado en todo momento, me sostiene y me alienta a perseverar. Es por ella que mis ideas se convierten en objetivos y de esta forma he logrado crecer como persona y como profesional. Con mucho amor, extendiendo esta dedicatoria a mi futura hija Nazli, quien desde ya me inspira a ser un excelente padre y aspiro poder brindarle oportunidades en su vida que la lleven a la felicidad.

AGRADECIMIENTOS

A mis tutores, Marco Ramírez y José Roberto Santamaría, por su guía y consejo durante el proceso y por compartir su amplio conocimiento con quienes aspiramos convertirnos en profesionales de excelencia.

A Carlos y Gaby, miembros de la Tribu Navi, por su gran cooperación y apertura durante las distintas fases de la investigación. Su compromiso con la organización los llevará a alcanzar sus objetivos.

EPÍGRAFE

“La principal razón para establecer un objetivo es en lo que te conviertes al lograrlo,
en lo que te conviertes siempre tendrá un valor mucho mayor”

Jim Rohn

ÍNDICE GENERAL

DEDICATORIA	I
AGRADECIMIENTOS	II
EPÍGRAFE.....	III
ÍNDICE GENERAL.....	IV
ÍNDICE DE FIGURAS.....	IX
ÍNDICE DE TABLAS	XI
ÍNDICE DE CUADROS.....	XII
LISTA DE ABREVIATURAS.....	XIII
RESUMEN	XIV
ABSTRACT	XV
INTRODUCCIÓN	16
CAPÍTULO 1 GENERALIDADES DE LA INVESTIGACIÓN	17
1.1 MARCO DE REFERENCIA ORGANIZACIONAL.....	17
1.1.1 Dirección de Informática	17
1.1.2 Tribu Navi	18
1.1.3 Estructura organizacional y marco estratégico.....	19
1.1.4 Proyectos en la organización	22
1.2 PLANTEAMIENTO DEL PROBLEMA	27
1.2.1 Análisis del problema	27
1.2.2 Enunciado del problema.....	35
1.2.3 Efectos del problema.....	36
1.2.4 Análisis de causas del problema.....	36

1.2.5	<i>Análisis de causa raíz</i>	40
1.2.6	<i>Árbol de problemas</i>	42
1.3	JUSTIFICACIÓN DEL ESTUDIO	43
1.4	OBJETIVOS	48
1.4.1	<i>Objetivo general</i>	48
1.4.2	<i>Objetivos específicos</i>	48
1.5	ALCANCE Y LIMITACIONES.....	49
1.5.1	<i>Alcance</i>	49
1.5.2	<i>Limitaciones</i>	52
CAPÍTULO 2 MARCO TEÓRICO		53
2.1	MARCO DE TRABAJO SCRUM	53
2.1.1	<i>Pilares</i>	54
2.1.2	<i>Valores</i>	54
2.1.3	<i>Roles</i>	55
2.1.4	<i>Eventos</i>	56
2.1.5	<i>Artefactos</i>	57
2.2	GESTIÓN DE REQUISITOS.....	58
2.2.1	<i>Proceso de gestión de requisitos</i>	58
2.2.2	<i>Requisitos y el ciclo de vida del proyecto</i>	68
2.2.3	<i>Gestión de requisitos y gestión de proyectos</i>	69
2.3	ESPECIFICACIONES DE REQUISITOS DE SISTEMAS.....	70
2.3.1	<i>Generalidades</i>	70
2.3.2	<i>Proceso</i>	73
2.3.3	<i>Requisitos bien formulados</i>	75
2.3.4	<i>Desarrollo de requisitos</i>	78
2.4	ESPECIFICACIONES DE REQUISITOS DE SOFTWARE.....	81
2.4.1	<i>Componentes de una especificación de requisitos de software</i>	81

2.4.2	<i>Consideraciones para formular especificaciones de requisitos</i>	85
2.5	<i>SOFTWARE DE GESTIÓN JIRA</i>	88
2.5.1	<i>Scrum con Jira</i>	89
2.5.2	<i>Hojas de ruta</i>	90
2.5.3	<i>Confluence</i>	91
CAPÍTULO 3 MARCO METODOLÓGICO		92
3.1	<i>CATEGORÍAS DE LA INVESTIGACIÓN</i>	92
3.2	<i>ETAPA DE RECOLECCIÓN DE INFORMACIÓN</i>	95
3.2.1	<i>Sujetos de información</i>	95
3.2.2	<i>Fuentes de información</i>	96
3.2.3	<i>Técnicas de recopilación de datos</i>	98
3.2.4	<i>Instrumentos de recopilación de datos</i>	100
3.3	<i>ETAPA DE PROCESAMIENTO Y ANÁLISIS DE DATOS</i>	103
3.3.1	<i>Métodos para el procesamiento y análisis de datos</i>	104
CAPÍTULO 4 ANÁLISIS DE RESULTADOS		108
4.1	<i>PRÁCTICAS ACTUALES DE GESTIÓN DE REQUISITOS</i>	108
4.1.1	<i>Resultados de la revisión documental</i>	108
4.1.2	<i>Resultados del grupo focal</i>	111
4.1.3	<i>Resultados de la entrevista</i>	113
4.1.4	<i>Análisis de resultados</i>	113
4.1.5	<i>Flujograma de procesos actuales de gestión de requisitos</i>	117
4.1.6	<i>Inventario de prácticas actuales de gestión de requisitos</i>	119
4.2	<i>BUENAS PRÁCTICAS DE GESTIÓN DE REQUISITOS</i>	120
4.2.1	<i>Resultados de la revisión documental</i>	120
4.2.2	<i>Resultados del análisis de casos de referencia</i>	121
4.2.3	<i>Inventario de buenas prácticas de gestión de requisitos</i>	123
4.2.4	<i>Cuadro de análisis</i>	126

CAPÍTULO 5 PROPUESTA DE SOLUCIÓN.....	129
5.1 FLUJOGRAMA DE PROCESOS DE GESTIÓN DE REQUISITOS.....	129
5.2 GUÍA METODOLÓGICA PARA LA GESTIÓN DE REQUISITOS.....	132
5.2.1 <i>Introducción</i>	132
5.2.2 <i>Proceso de evaluación de la necesidad</i>	132
5.2.3 <i>Proceso de planificación de la gestión de requisitos</i>	136
5.2.4 <i>Proceso de obtención de los requisitos</i>	139
5.2.5 <i>Proceso de análisis de los requisitos</i>	144
5.2.6 <i>Proceso de evaluación de la solución</i>	148
5.2.7 <i>Proceso de monitoreo y control de los requisitos</i>	151
5.3 DICCIONARIO DE TÉRMINOS PARA LA GESTIÓN DE REQUISITOS.....	153
5.4 ESTRATEGIA DE IMPLEMENTACIÓN DEL MÉTODO DE GESTIÓN DE REQUISITOS.....	153
5.4.1 <i>Fases y actividades de la implementación</i>	153
5.4.2 <i>Presupuesto</i>	157
5.4.3 <i>Cronograma</i>	158
CAPÍTULO 6 CONCLUSIONES Y RECOMENDACIONES	159
6.1 CONCLUSIONES	159
6.2 RECOMENDACIONES.....	161
CAPÍTULO 7 REFERENCIAS BIBLIOGRÁFICAS	162
CAPÍTULO 8 APÉNDICES.....	164
8.1 APÉNDICE A. CUESTIONARIO DE DIAGNÓSTICO.....	164
8.2 APÉNDICE B. GUÍA DE ENTREVISTA.....	172
8.3 APÉNDICE C. FORMATO DE FICHA DE SÍNTESIS	173
8.4 APÉNDICE D. FORMATO DE INVENTARIO	174
8.5 APÉNDICE E. FORMATO DE LISTA DE VERIFICACIÓN	175
8.6 APÉNDICE F. FICHAS DE SÍNTESIS DEL MANUAL OPERATIVO <i>SCRUM</i>	177

8.7	APÉNDICE G. RESULTADOS DE LISTA DE VERIFICACIÓN.....	181
8.8	APÉNDICE H. RESULTADOS DE LA ENTREVISTA	199
8.9	APÉNDICE I. FICHAS DE SÍNTESIS DE PRÁCTICAS DE REFERENCIA - U.S. DEPARTMENT OF TRANSPORTATION 201	
8.10	APÉNDICE J. FICHAS DE SÍNTESIS DE PRÁCTICAS DE REFERENCIA - NASA.....	205
8.11	APÉNDICE K. FICHAS DE SÍNTESIS DE PRÁCTICAS DE REFERENCIA – INFOSYS.....	218
8.12	APÉNDICE L. FICHAS DE SÍNTESIS DE BUENAS PRÁCTICAS.....	222
8.13	APÉNDICE M. PLANTILLA DEL PLAN DE GESTIÓN DE REQUISITOS.....	235
8.14	APÉNDICE N. PLANTILLA DEL DOCUMENTO DE REQUISITOS DEL PRODUCTO (PRD).....	236
8.15	APÉNDICE O. DICCIONARIO DE TÉRMINOS PARA LA GESTIÓN DE REQUISITOS	238

ÍNDICE DE FIGURAS

FIGURA 1.1. ESTRUCTURACIÓN DE LA ENTIDAD FINANCIERA	18
FIGURA 1.2. ESTRUCTURA ORGANIZACIONAL DE LA TRIBU NAVI	19
FIGURA 1.3. ESTRUCTURA DE SCRUM OF SCRUMS Y METASCRUM	20
FIGURA 1.4. CICLO ITERATIVO DE PROYECTOS DE LA TRIBU NAVI	22
FIGURA 1.5. METASCRUM EN LA TRIBU NAVI	26
FIGURA 1.6. GRÁFICOS DE RENDIMIENTO DE EQUIPOS POR SPRINT	34
FIGURA 1.7. DOCUMENTACIÓN DE REQUISITOS	41
FIGURA 1.8. ÁRBOL DE PROBLEMAS	42
FIGURA 1.9. ESFUERZOS PARA LA MEJORA DE LA GESTIÓN DE LOS REQUISITOS	44
FIGURA 1.10. COMPARATIVA DE RESULTADOS ORGANIZACIONALES CON Y SIN PROCESOS FORMALES	45
FIGURA 1.11. CAUSAS MITIGADAS CON LA SOLUCIÓN	46
FIGURA 1.12. ESTRUCTURA DE DESGLOSE DE LOS ENTREGABLES	50
FIGURA 2.1. PROCESO DE GESTIÓN DE REQUISITOS	59
FIGURA 2.2. GESTIÓN DE REQUISITOS SEGÚN EL CICLO DE VIDA DEL PROYECTO	69
FIGURA 2.3. CONTEXTO DE DESARROLLO DE ESPECIFICACIONES DE REQUISITOS DE SISTEMAS	73
FIGURA 2.4. PROCESO DE DESARROLLO DE ESPECIFICACIONES DE REQUISITOS DE SISTEMAS	79
FIGURA 4.1. FLUJOGRAMA DE PROCESOS DE GESTIÓN DE REQUISITOS	118
FIGURA 5.1. FLUJOGRAMA GENERAL DE PROCESOS DE LA GESTIÓN DE REQUISITOS	130
FIGURA 5.2. FLUJOGRAMA DETALLADO DE PROCESOS DE LA GESTIÓN DE REQUISITOS	131
FIGURA 5.3. PROCESO DE EVALUACIÓN DE LA NECESIDAD	133
FIGURA 5.4. PLANTILLA DEL ENUNCIADO DE LA NECESIDAD	135
FIGURA 5.5. PLANTILLA DE LA MATRIZ DE INTERESADOS Y EXPECTATIVAS	135
FIGURA 5.6. PROCESO DE PLANIFICACIÓN DE LA GESTIÓN DE REQUISITOS	136
FIGURA 5.7. PROCESO DE OBTENCIÓN DE LOS REQUISITOS	140
FIGURA 5.8. PROCESO DE ANÁLISIS DE LOS REQUISITOS	144

FIGURA 5.9. PROCESO DE EVALUACIÓN DE LA SOLUCIÓN.....	149
FIGURA 5.10. PROCESO DE MONITOREO Y CONTROL DE REQUISITOS	151
FIGURA 5.11. CRONOGRAMA DE IMPLEMENTACIÓN.....	158

ÍNDICE DE TABLAS

TABLA 1.1. PORCENTAJE DE TRABAJO NO COMPLETADO POR SPRINT – EQUIPO OMA	28
TABLA 1.2. PORCENTAJE DE TRABAJO NO COMPLETADO POR PROYECTO - EQUIPO OMA	29
TABLA 1.3. PORCENTAJE DE TRABAJO NO COMPLETADO POR SPRINT – EQUIPO TAW	30
TABLA 1.4. PORCENTAJE DE TRABAJO NO COMPLETADO POR PROYECTO - EQUIPO TAW	30
TABLA 1.5. PORCENTAJE DE TRABAJO NO COMPLETADO POR SPRINT – EQUIPO KEK	31
TABLA 1.6. PORCENTAJE DE TRABAJO NO COMPLETADO POR PROYECTO - EQUIPO KEK	32
TABLA 1.7. PORCENTAJE DE TRABAJO NO COMPLETADO POR SPRINT – EQUIPO IKN	33
TABLA 1.8. PORCENTAJE DE TRABAJO NO COMPLETADO POR PROYECTO - EQUIPO IKN	33
TABLA 1.9. CASOS ESPECÍFICOS QUE EVIDENCIAN LAS CAUSAS DEL TRABAJO NO COMPLETADO	38
TABLA 1.10. PROYECCIÓN DE LAS ENTREGAS DE PROYECTOS A TIEMPO – EQUIPO OMA	47
TABLA 4.1. PRÁCTICAS ACTUALES DE DOCUMENTACIÓN DE REQUISITOS	109
TABLA 4.2. PRÁCTICAS ACTUALES DE GESTIÓN DE REQUISITOS	111
TABLA 4.3. PRÁCTICAS ACTUALES DE FORMULACIÓN DE REQUISITOS	112
TABLA 4.4. COMPETENCIA EN GESTIÓN DE REQUISITOS	113
TABLA 5.1. PRESUPUESTO DE LA IMPLEMENTACIÓN	157

ÍNDICE DE CUADROS

CUADRO 1.1. PROYECTOS EJECUTADOS POR LA TRIBU NAVI.....	25
CUADRO 1.2. CAUSAS DEL TRABAJO NO COMPLETADO.....	37
CUADRO 3.1. CATEGORÍAS DE LA INVESTIGACIÓN.....	94
CUADRO 3.2. SUJETOS DE INFORMACIÓN.....	95
CUADRO 3.3. FUENTES DE INFORMACIÓN.....	97
CUADRO 3.4. MÉTODOS PARA EL PROCESAMIENTO Y ANÁLISIS DE LOS DATOS	107
CUADRO 4.1. INVENTARIO DE PRÁCTICAS ACTUALES DE GESTIÓN DE REQUISITOS.....	119
CUADRO 4.2. INVENTARIO DE BUENAS PRÁCTICAS DE GESTIÓN DE REQUISITOS	123
CUADRO 4.3. CUADRO DE ANÁLISIS DE PRÁCTICAS ACTUALES Y BUENAS PRÁCTICAS SELECCIONADAS	126

LISTA DE ABREVIATURAS

IEEE: *Institute of Electrical and Electronics Engineers.*

PMBok®: *Project Management Body of Knowledge.*

PMI®: *Project Management Institute.*

PRD: *Product Requirements Document.*

RESUMEN

La Tribu Navi está conformada por equipos multidisciplinarios *Scrum* dedicados al desarrollo ágil de software para una entidad financiera con sede en Costa Rica. En enero de 2021, la Tribu Navi descubrió que el trabajo comprometido en cada iteración de desarrollo difería del trabajo realmente completado, afectando así sus tiempos de entrega e incumpliendo con los compromisos adquiridos con los interesados.

Al realizar un estudio a profundidad, se determinaron las causas más frecuentes que provocaban que el trabajo no fuese completado y se llegó a la conclusión de que dichas causas tenían una relación directa con la gestión de requisitos de los proyectos. Para alcanzar una solución a esta problemática, se realizaron grupos focales, entrevistas y revisión documental que permitieron identificar las prácticas de gestión de requisitos adoptadas por la Tribu Navi y los elementos de la problemática relacionados. Posteriormente, se realizó revisión de literatura y análisis de casos de referencia de organizaciones de prestigio que permitieron identificar prácticas recomendadas para la mitigación de cada elemento de la problemática.

Con estos insumos fue posible proponer un método estándar de gestión de requisitos para la Tribu Navi, el cual se conforma de un flujograma de procesos, una guía metodológica detallada y un diccionario de términos para la gestión de requisitos. Además, se desarrolló una estrategia de implementación que facilita la adopción del método y la adaptación al cambio en la Tribu Navi, fortaleciendo sus competencias individuales para gestionar adecuadamente los requisitos de sus proyectos. Una vez implementado, el método solventará la problemática mediante la incorporación de prácticas estandarizadas de gestión de requisitos acordes al marco de trabajo *Scrum*.

Palabras Clave: gestión de requisitos, administración de proyectos, *Scrum*, buenas prácticas, desarrollo de software.

ABSTRACT

The Navi Tribe is made up of multidisciplinary Scrum teams dedicated to agile software development for a bank based in Costa Rica. In January 2021, the Navi Tribe discovered that the work committed in each iteration of development differed from the work completed, thus affecting their delivery time, and failing to meet commitments made with stakeholders.

When conducting an in-depth study, the most frequent causes that caused the work not to be completed were determined and it was concluded that these causes had a direct relationship with the management of project requirements. To reach a solution to this problem, focus groups, interviews and documentary reviews were carried out that made it possible to identify the current requirements management practices adopted by the Navi Tribe and the related elements of the problem. Subsequently, documentary reviews of different world-renowned authors and analysis of reference cases from prestigious organizations were carried out, which made it possible to identify good practices for the mitigation of each of the elements of the problem.

With these inputs it was possible to propose a standardized requirements management method for the Navi Tribe, which is made up of a process flow chart, a detailed methodological guide, and a dictionary of terms for requirements management. In addition, an implementation strategy was developed that facilitates the adoption of the method and adaptation to change in the Navi Tribe, strengthening their current skills required to adequately manage the requirements of their projects. Once implemented, the method will solve the problem by incorporating standardized requirements management practices according to the Scrum framework.

Keywords: Requirements Management, project management, Scrum, good practices, software development.

INTRODUCCIÓN

Con la rápida propagación de las metodologías ágiles en las organizaciones dedicadas al desarrollo de *software*, surgieron nuevos retos que ponen a prueba la capacidad de adaptación de las metodologías de gestión de proyectos. Los métodos ágiles invitan a dedicar menos tiempo en la definición inicial de los requisitos del producto para invertirlo en la inspección y adaptación de forma iterativa. Ahora bien, muchas organizaciones malinterpretan la agilidad y caen en la informalidad, omitiendo o ignorando la importancia que tiene la definición de procesos claros de gestión de requisitos en sus proyectos.

La correcta gestión de los requisitos permite a las organizaciones alcanzar los objetivos establecidos de sus proyectos, incluyendo el cumplimiento de los plazos de entrega y el aumento de la satisfacción de los interesados. Debido a lo anterior, este escrito tiene como objetivo responder a la problemática identificada en la Tribu Navi, mediante la propuesta de un método de gestión de requisitos de proyectos y su estrategia de implementación.

En el capítulo 1, se exponen el marco de referencia organizacional, el planteamiento del problema, la justificación, los objetivos, el alcance y las limitaciones del estudio. En el capítulo 2, se detallan los conceptos teóricos relacionados a la problemática y a la solución propuesta que fundamentan la investigación. En el capítulo 3, se expone la metodología utilizada para desarrollar la investigación, los entregables y la solución a la problemática. En el capítulo 4, se presentan y analizan los resultados obtenidos de la investigación, según lo definido en el marco metodológico. En el capítulo 5, se desarrolla la propuesta de solución de la problemática, la cual consiste en un método de gestión de requisitos de proyectos para la Tribu Navi. En el capítulo 6, se exponen las conclusiones y recomendaciones del estudio. Finalmente, en los capítulos 7 y 8 se muestran las referencias bibliográficas y los apéndices, respectivamente.

Capítulo 1 Generalidades de la investigación

En este capítulo se presenta el marco de referencia de la organización donde se elaboró el presente estudio, así como el planteamiento del problema y la justificación. Además, se exponen los objetivos propuestos, el alcance de la investigación y las limitaciones encontradas.

1.1 Marco de referencia organizacional

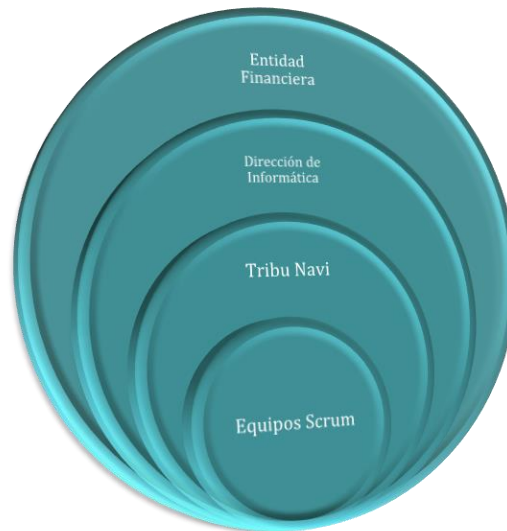
El presente estudio fue elaborado en la Dirección de Informática de una entidad financiera transnacional que brinda sus servicios en Costa Rica desde hace más de 30 años. El nombre de la entidad financiera no será mencionado por aspectos de confidencialidad.

1.1.1 Dirección de Informática

La Dirección de Informática funge como una unidad autónoma e independiente de las operaciones financieras y está constituida como sociedad anónima en Costa Rica. Esta se dedica al desarrollo de productos y servicios tecnológicos para satisfacer las necesidades de las operaciones de la entidad financiera. Asimismo, cuenta con más de 500 colaboradores y una experiencia de más de 20 años en el desarrollo de *software* para servicios financieros.

Como se muestra en la Figura 1.1, la Dirección de Informática se divide en unidades conocidas como tribus, encargadas de desarrollar soluciones tecnológicas para servicios financieros específicos. El presente estudio fue realizado en la Tribu Navi.

Figura 1.1. Estructuración de la entidad financiera



Nota: Adaptado de *Organigrama de la entidad financiera*.

1.1.2 Tribu Navi

La Tribu Navi está conformada por más de 30 colaboradores tanto de negocio como de tecnologías de la información y utiliza el marco de trabajo *Scrum* para la gestión de sus proyectos. La tribu cuenta con cuatro equipos *Scrum* de desarrollo de software encabezados por un jefe administrativo que se desempeña como *Chief Scrum Master* y por un jefe de negocio que funge como *Chief Product Owner* y patrocinador de los proyectos.

La Tribu Navi fue pionera en la adopción del marco de trabajo *Scrum* en la organización y cuenta con más de cuatro años de experiencia desarrollando productos y servicios de software bajo esta metodología. Los colaboradores de la Tribu Navi tienen entre dos y veinticinco años de pertenecer a la organización, y sus perfiles profesionales son: analista de negocio, líder de proyectos (*Product Owner*), *Scrum Master*, ingeniero de software e ingeniero de redes.

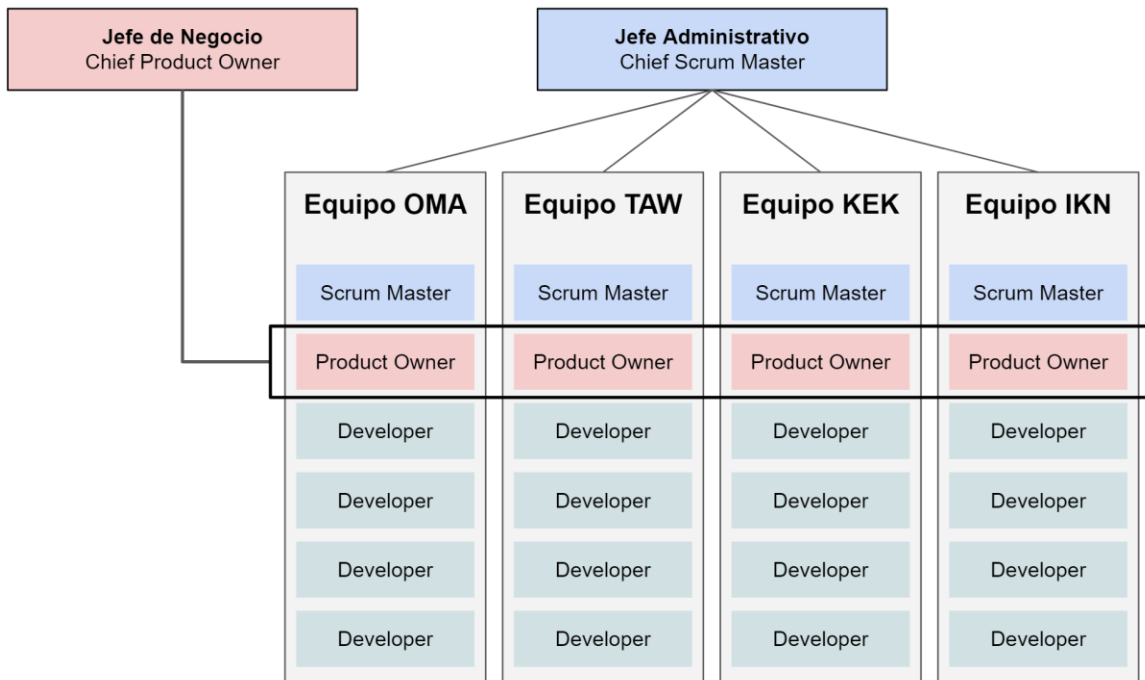
1.1.3 Estructura organizacional y marco estratégico.

A continuación, se detalla la estructura organizacional y el marco estratégico de la Tribu Navi.

1.1.3.1 Estructura Organizacional.

La Tribu Navi cuenta con una estructura organizacional acorde al marco de trabajo *Scrum* para la ejecución de proyectos de desarrollo de software. Como se muestra en la Figura 1.2, la Tribu Navi es liderada por un jefe administrativo a cargo de cuatro equipos *Scrum* y un jefe de negocio quien lidera a los *Product Owners* y la cartera de proyectos. Cada uno de los equipos se compone de un *Scrum Master*, un *Product Owner* y varios *developers*.

Figura 1.2. Estructura organizacional de la Tribu Navi

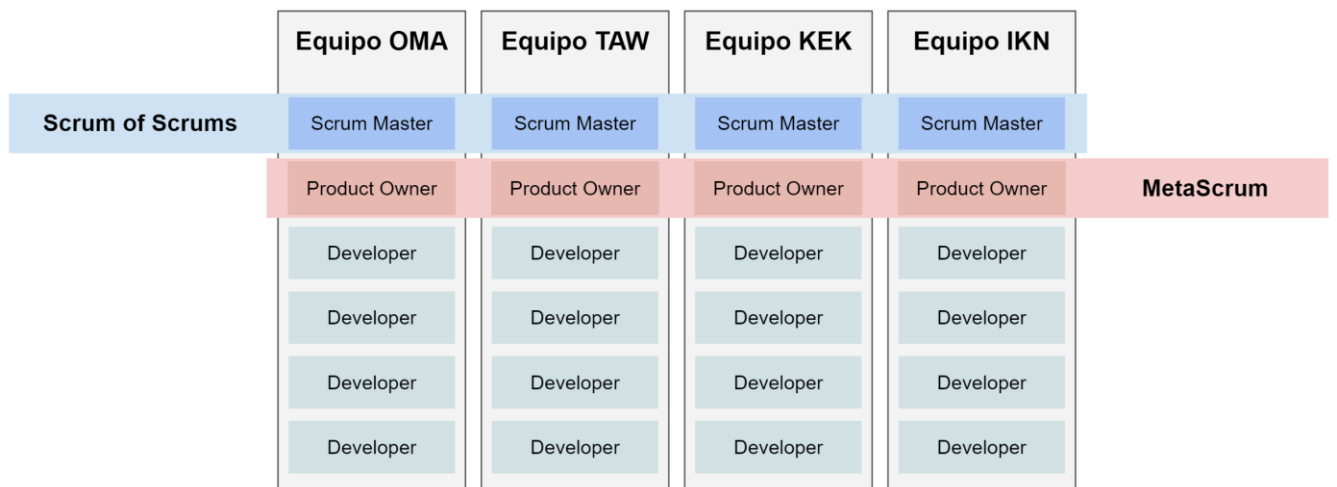


Nota: Adaptado de *Organigrama de la Tribu Navi*

Como se muestra en la Figura 1.3, la estructura organizacional también define dos equipos de trabajo adicional:

- *Scrum of Scrums*: compuesto por los *Scrum Master* de cada equipo *Scrum* y el *Chief Scrum Master*; y tiene como objetivo mantener y mejorar los procesos de *Scrum* de la tribu para el correcto desempeño de los proyectos.
- *MetaScrum*: compuesto por los *Product Owners* de cada equipo *Scrum* y el *Chief Product Owner*; y tiene como objetivo gestionar la cartera de proyectos de la Tribu Navi.

Figura 1.3. Estructura de Scrum of Scrums y MetaScrum



Nota: Adaptado de Organigrama de la Tribu Navi

1.1.3.2 Marco estratégico.

La Tribu Navi cuenta con un marco estratégico propio, alineado a la misión y visión organizacional. Este marco establece la siguiente visión:

- Buscar el crecimiento de cada miembro de la tribu.
- Mejorar continuamente todos los procesos y productos generados por la tribu.
- Buscar la innovación, generando espacios de investigación que incentiven la creatividad de la tribu.
- Fomentar el trabajo en equipo manteniendo los valores de apoyo y confianza que permita cumplir los objetivos de la tribu.
- Mejorar la confianza de los clientes mediante el correcto y continuo uso de las metodologías de trabajo ágiles.
- Generar tiempos de entrega a mercado que estén por encima de los competidores.
- Fomentar la creación de espacios de comunicación claros y efectivos entre todos los miembros de la tribu.

Además, la Tribu Navi cuenta con los siguientes objetivos:

- Sentirse orgullosos por los productos y servicios que son brindados a los clientes y ser ejemplo de excelencia para las diferentes áreas de la empresa.
- Crecer en penetración basados en la creación de medios de pago flexibles y atractivos para los distintos segmentos de mercado, dando prioridad a los que no cuentan con los servicios de la empresa.
- Ser líderes del mercado por medio de la innovación y con el fin de generar una alta dependencia en los clientes hacia nuestros productos.

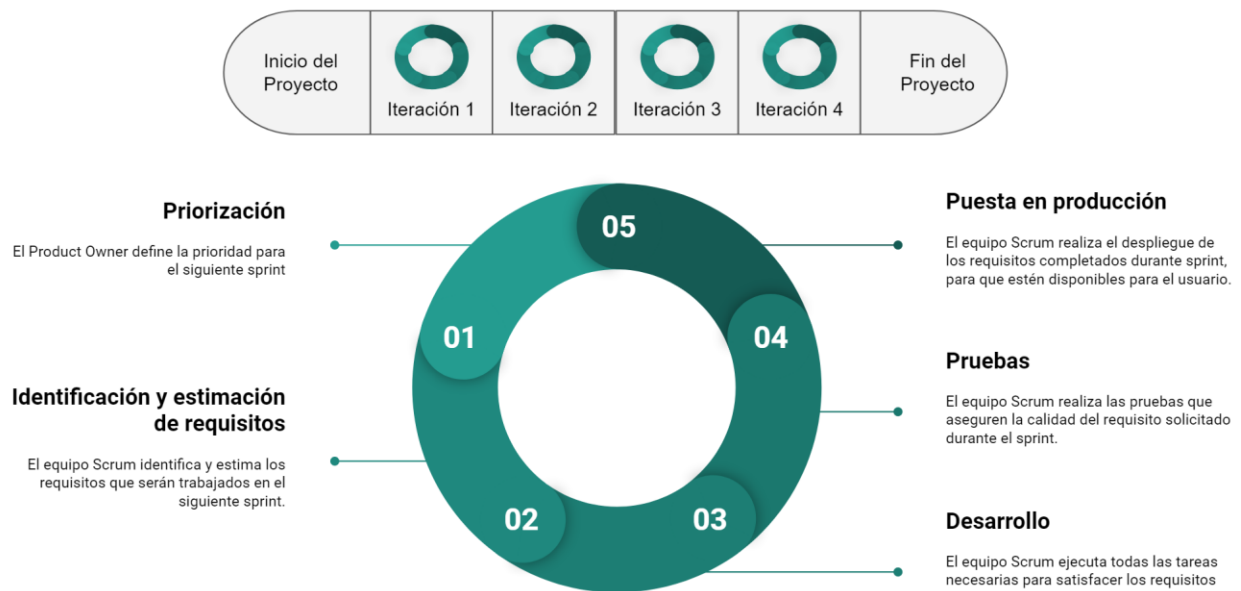
1.1.4 Proyectos en la organización.

A continuación, se detallan las características de los proyectos en la Tribu Navi.

1.1.4.1 Desarrollo iterativo e incremental de proyectos.

La Tribu Navi se especializa en el desarrollo ágil de proyectos de software para productos y servicios financieros, bajo el marco de trabajo *Scrum* y los principios ágiles. Además, los propósitos se desarrollan de forma iterativa e incremental como se muestra en la Figura 1.4. El ciclo de vida de los proyectos se extiende usualmente entre tres y seis meses y contempla la ejecución de una serie de iteraciones de dos semanas conocidas como *sprints*. En cada sprint el producto o servicio se construye según las especificaciones de requisitos y la retroalimentación de los usuarios finales, el patrocinador y el *Product Owner*. Cada iteración se conforma de cinco procesos: priorización, identificación y estimación de requisitos; desarrollo, pruebas y puesta en producción.

Figura 1.4. Ciclo iterativo de proyectos de la Tribu Navi



Nota: Adaptado de la *Metodología de la Tribu Navi*

1.1.4.2 Alcance de los proyectos.

El alcance de los proyectos en la Tribu Navi se mantiene abierto a cambios durante las iteraciones y no se establece un plan de gestión del alcance al inicio de los proyectos. Durante cada iteración, los equipos *Scrum* instauran el alcance de la esta, en coordinación con su *Product Owner*, basándose en el valor de negocio que generará. A la vez, la eficacia del producto o servicio es nuevamente definido, por lo que los equipos *Scrum* deben estar en constante planificación del alcance específico de sus sprints.

1.1.4.3 Costo de los proyectos.

Los proyectos de la Tribu Navi no cuentan con un presupuesto inicial ni acuerdos contractuales, ya que, al ser una unidad dedicada al desarrollo de proyectos de software, sus costos no son establecidos por proyecto, sino considerados constantes. Esto implica que la Tribu Navi no asigna recurso humano a los proyectos, sino que los determina a los equipos *Scrum*. Se estima que el costo promedio de operación de un equipo *Scrum* conformado por seis personas ronda los cuatro millones de colones por iteración de 15 días.

1.1.4.4 Clasificación de proyectos.

Los proyectos en la Tribu Navi se clasifican en:

- Proyectos de nuevos productos o servicios. Estos consisten en el desarrollo de nuevos productos o servicios financieros que serán desplegados por las áreas operativas para satisfacer una oportunidad de negocio.
- Proyectos de mejora de productos o servicios existentes. Estos consisten en la modificación de los productos o servicios financieros existentes para incorporar

nuevas funcionalidades, realizar cambios en las actuales o corregir algún comportamiento no deseado.

- Proyectos para el cumplimiento regulatorio. Estos consisten en el desarrollo de nuevos productos o la modificación de mercados actuales para el cumplimiento de normas, reglamentos o leyes que impactan la operativa financiera. Además, se clasifican como proyectos para el cumplimiento regulatorio todas aquellas iniciativas relacionadas con seguridad de la información.
- Proyectos para mejora de la operativa. Estos consisten en el desarrollo de productos o servicios que maximicen el rendimiento o que disminuyen los costos de las áreas operativas.

En el Cuadro 1.1 se muestran los proyectos más relevantes trabajados por la Tribu Navi, desde agosto de 2020 a marzo de 2021. Es posible observar que predominaron los proyectos relacionados al desarrollo de nuevos productos o servicios.

Cuadro 1.1. Proyectos ejecutados por la Tribu Navi

Equipo Asignado	Proyecto	Clasificación
OMA	SafeKey AX	Nuevos productos o servicios
OMA	Tarjetas Clave	Nuevos productos o servicios
OMA	Segregación	Nuevos productos o servicios
OMA	Migración WAS 8.5	Mejora de productos o servicios
OMA	3DS 2.0	Cumplimiento regulatorio
TAW	Migración Automática	Nuevos productos o servicios
TAW	Device Fingerprint	Nuevos productos o servicios
TAW	AVP SILP Centralizado	Mejora de la operativa
TAW	AWS Voucher	Nuevos productos o servicios
TAW	Ajustes SILP Centralizado	Mejora de la operativa
KEK	AIS para adm. de pinpad	Mejora de la operativa
KEK	Pases AIS CSP	Mejora de la operativa
KEK	Check in - Check Out PAN	Nuevos productos o servicios
KEK	CSP - PIN ONLINE	Cumplimiento regulatorio
KEK	Adm remota fase III	Mejora de la operativa
IKN	Cliente Web Mobile	Nuevos productos o servicios
IKN	Cliente Web CNB	Nuevos productos o servicios
IKN	Cliente Web – Sec. Breach	Cumplimiento regulatorio
IKN	Retiro dinero sin tarjeta	Mejora de productos o servicios
IKN	CSP CNB a Centralizado	Mejora de la operativa

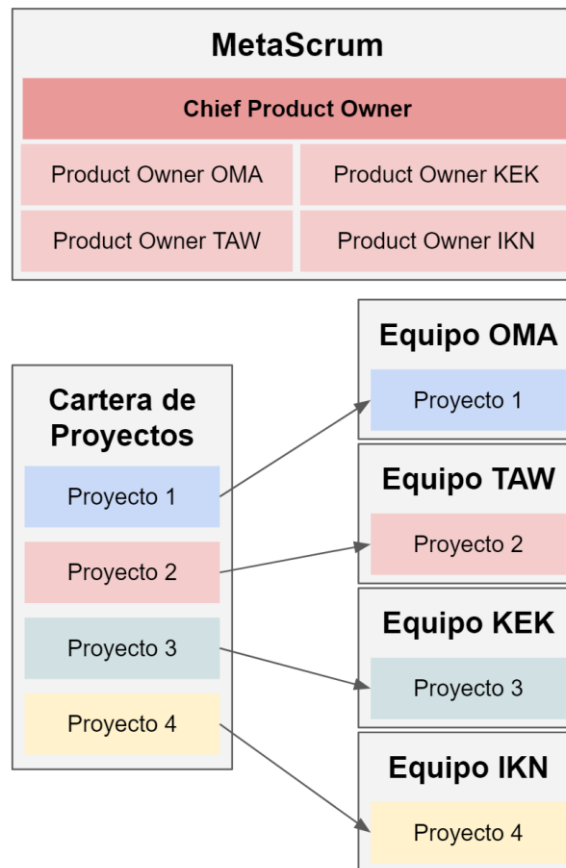
Nota: Adaptado de Reportes Jira.

1.1.4.5 Priorización de proyectos.

El *MetaScrum* es el equipo encargado de gestionar la cartera de proyectos de la Tribu Navi. La cartera se compone de todos aquellos proyectos que han sido solicitados por las áreas de operaciones financieras y están en cola de espera para ser ejecutados. El *Chief Product Owner* en conjunto con los *Product Owners* de cada equipo *Scrum* se encargan de establecer prioridades a los proyectos de la cartera y

asignarlos tomando en cuenta el tipo de proyecto, la experiencia y la disponibilidad de los equipos *Scrum*, como se muestra en la Figura 1.5. Estos procesos de priorización y asignación de proyectos se realizan una vez por *sprint*, de forma que la Tribu Navi cuente con la agilidad requerida para reaccionar y adaptarse a las necesidades del negocio.

Figura 1.5. MetaScrum en la Tribu Navi



Nota: Adaptado de la Metodología de la Tribu Navi

1.2 Planteamiento del problema

A continuación, se presenta el planteamiento del problema.

1.2.1 Análisis del problema

En enero de 2021, se implementó un indicador de medición de desempeño grupal para los equipos *Scrum* OMA, TAW, KEK e IKN que conforman la Tribu Navi. El indicador consistió en la medición del trabajo no completado durante el *sprint*, el cual se considera un incumplimiento de los compromisos adquiridos por los equipos *Scrum* y una inadecuada planificación del *sprint*.

Debido a que en los primeros tres meses desde la implementación del indicador no se obtuvieron los resultados deseados, en abril de 2021, los *Scrum Master* de la Tribu Navi realizaron un estudio del rendimiento de los cuatro equipos de desarrollo de software. En el estudio, se analizó el trabajo realizado por los equipos *Scrum* durante 12 *sprints* comprendidos entre agosto de 2020 y marzo de 2021. Se excluyeron del estudio dos *sprints* que coincidieron con las vacaciones de fin de año, ya que más del 70% de las personas estuvieron de vacaciones obligatorias.

El estudio consistió en el análisis de los requisitos documentados en el software Jira para determinar el porcentaje de trabajo no completado durante los *sprints* y su efecto en los cinco proyectos más significativos de cada equipo *Scrum*. Se utilizaron los valores de esfuerzo medidos en puntos para medir el trabajo comprometido y el trabajo no completado en cada *sprint*. La utilización de puntos en lugar de tiempo para medir el esfuerzo es una práctica adoptada por los equipos *Scrum* desde su creación. El estudio partió de la premisa de que el trabajo comprometido por el equipo *Scrum* durante el *Sprint Planning* debe ser completado en su totalidad durante el *sprint*.

En la Tabla 1.1, se observa que el equipo OMA se comprometió a realizar esfuerzos comprendidos entre los 18 y los 38 puntos por *sprint*, para un total de 358 puntos. Del total de trabajo comprometido por el equipo, 76 puntos no lograron completarse durante el *sprint*, lo que

corresponde al 21%. Cabe destacar que el equipo OMA logró completar todo lo comprometido solamente en tres de los 12 *sprints* analizados, lo que significa que el equipo dejó trabajo incompleto en 75% de los *sprints* que planificó y ejecutó.

Tabla 1.1. Porcentaje de trabajo no completado por Sprint – Equipo OMA

Sprint	Trabajo comprometido	Trabajo no completado	
100	21	8	38%
101	28	15	54%
102	27	0	0%
103	38	9	24%
104	29	14	48%
105	30	0	0%
106	30	0	0%
107	29	4	14%
108	18	1	6%
111	37	5	13%
112	34	9	26%
113	36	11	30%
TOTAL	358	76	Promedio: 21%

Nota: Adaptado de Reportes Jira.

El porcentaje de trabajo no completado por *sprint* tuvo un impacto directo en los proyectos ejecutados por el equipo OMA. Como se muestra en la Tabla 1.2, existieron porcentajes significativos (18% - 58%) de trabajo que no logró ser completado en el tiempo que fue comprometido, lo que obligó al *Product Owner* a replanificar el compromiso para realizarse en *sprints* futuros. Ninguno de los proyectos logró finalizar sin complicaciones, a pesar de que dos de ellos requirieron esfuerzos relativamente bajos (19 puntos).

Tabla 1.2. Porcentaje de trabajo no completado por proyecto - Equipo OMA

Proyecto	Trabajo comprometido	Trabajo no completado	
SafeKey AX	153	28	18%
Tarjetas Clave	62	11	17%
Segregación	50	9	18%
Migración WAS 8.5	19	5	26%
3DS 2.0	19	11	58%

Nota: Adaptado de Reportes Jira.

Los datos de la

Tabla 1.3 muestran que el equipo TAW comprometió entre 19 y 38 puntos por *sprint*, de forma muy similar a como lo realizó el equipo OMA. De los 350 puntos totales comprometidos, no logró completar 71 de ellos, lo que equivale al 20% de trabajo no completado. El equipo TAW logró consumir todo lo comprometido en cuatro de los 12 *sprints* analizados, lo que representa que en un 67% de los *sprints* no logró finalizar el trabajo expuesto en su totalidad.

Como se muestra en la Tabla 1.4, la problemática provocó retrasos en tres de los proyectos, los cuales tuvieron 30%, 52% y 38% de su trabajo no completado a tiempo, mientras que dos de los proyectos lograron completarse sin dificultades. Los porcentajes de labor no completada a tiempo, colocan a TAW como el equipo con mayor afectación en sus proyectos en comparación con el resto de los equipos.

Tabla 1.3. Porcentaje de trabajo no completado por Sprint – Equipo TAW

Sprint	Trabajo comprometido	Trabajo no completado	
100	19	0	0%
101	26	8	31%
102	35	16	46%
103	33	5	15%
104	30	9	30%
105	32	0	0%
106	33	12	36%
107	21	6	28%
108	26	5	19%
111	31	10	32%
112	26	0	0%
113	38	0	0%
TOTAL	350	71	Promedio: 20%

Nota: Adaptado de Reportes Jira.

Tabla 1.4. Porcentaje de trabajo no completado por proyecto - Equipo TAW

Proyecto	Trabajo comprometido	Trabajo no completado	
Migración Automática	145	43	30%
Device Fingerprint	51	0	0%
AVP SILP Centralizado	40	21	52%
AWS Voucher	39	0	0%
Ajustes SILP Centralizado	13	5	38%

Nota: Adaptado de Reportes Jira.

Según la Tabla 1.5, el equipo KEK comprometió entre 24 y 47 puntos por *sprint*, lo que resultó en 424 puntos totales comprometidos. Del total de 424 puntos, 18 no lograron completarse, lo que equivale al 4%. Este valor es muy inferior al 21% y 20% obtenidos por los equipos OMA y TAW respectivamente, lo que significa que KEK fue capaz de completar mayor porcentaje de su trabajo. Es notable que el equipo KEK logró completar todo lo comprometido en nueve de los 12 *sprints* analizados, lo que representa que en un 25% no logró completar la tarea en su totalidad.

Tabla 1.5. Porcentaje de trabajo no completado por Sprint – Equipo KEK

Sprint	Trabajo comprometido	Trabajo no completado	
100	35	0	0%
101	31	0	0%
102	36	0	0%
103	37	0	0%
104	30	0	0%
105	41	5	12%
106	47	8	17%
107	41	0	0%
108	35	5	14%
111	30	0	0%
112	37	0	0%
113	24	0	0%
TOTAL	424	18	Promedio: 4%

Nota: Adaptado de Reportes Jira.

Como se muestra en la Figura 1.6, dos de los proyectos ejecutados por el equipo KEK presentaron retrasos y tres de ellos lograron su finalización sin contratiempos.

Tabla 1.6. Porcentaje de trabajo no completado por proyecto - Equipo KEK

Proyecto	Trabajo comprometido	Trabajo no completado	
AIS para adm. de pinpad	71	0	0%
Pases AIS CSP	40	5	13%
Check in - Check Out PAN	43	5	12%
CSP - PIN ONLINE	27	0	0%
Adm. remota fase III	18	0	0%

Nota: Adaptado de *Reportes Jira*.

En la Tabla 1.7, se observa que el equipo IKN comprometió entre 35 y 63 puntos por *sprint*, valores superiores a los equipos previamente analizados. Esto no significó que el equipo IKN fuese más capaz que el resto de los equipos, sino que, simplemente utilizó una escala de referencia distinta para estimar su esfuerzo. Del total de 575 puntos, 31 no lograron completarse, lo que equivale al 5%. Además, logró completar todo lo comprometido en ocho de los 12 *sprints* analizados, lo que representa que un 33% de los *sprints* no logró completar el trabajo comprometido en su totalidad.

En la Tabla 1.8, se observa que el trabajo no completado provocó retrasos en cuatro de los cinco proyectos ejecutados por IKN, especialmente en el Retiro dinero sin tarjeta, el cual presentó un 38%.

Tabla 1.7. Porcentaje de trabajo no completado por Sprint – Equipo IKN

Sprint	Trabajo comprometido	Trabajo no completado	
100	60	0	0%
101	50	5	10%
102	46	0	0%
103	45	0	0%
104	40	5	12%
105	38	0	0%
106	54	0	0%
107	39	8	20%
108	35	0	0%
111	54	0	0%
112	63	13	21%
113	52	0	0%
TOTAL	575	31	Promedio: 5%

Nota: Adaptado de Reportes Jira.

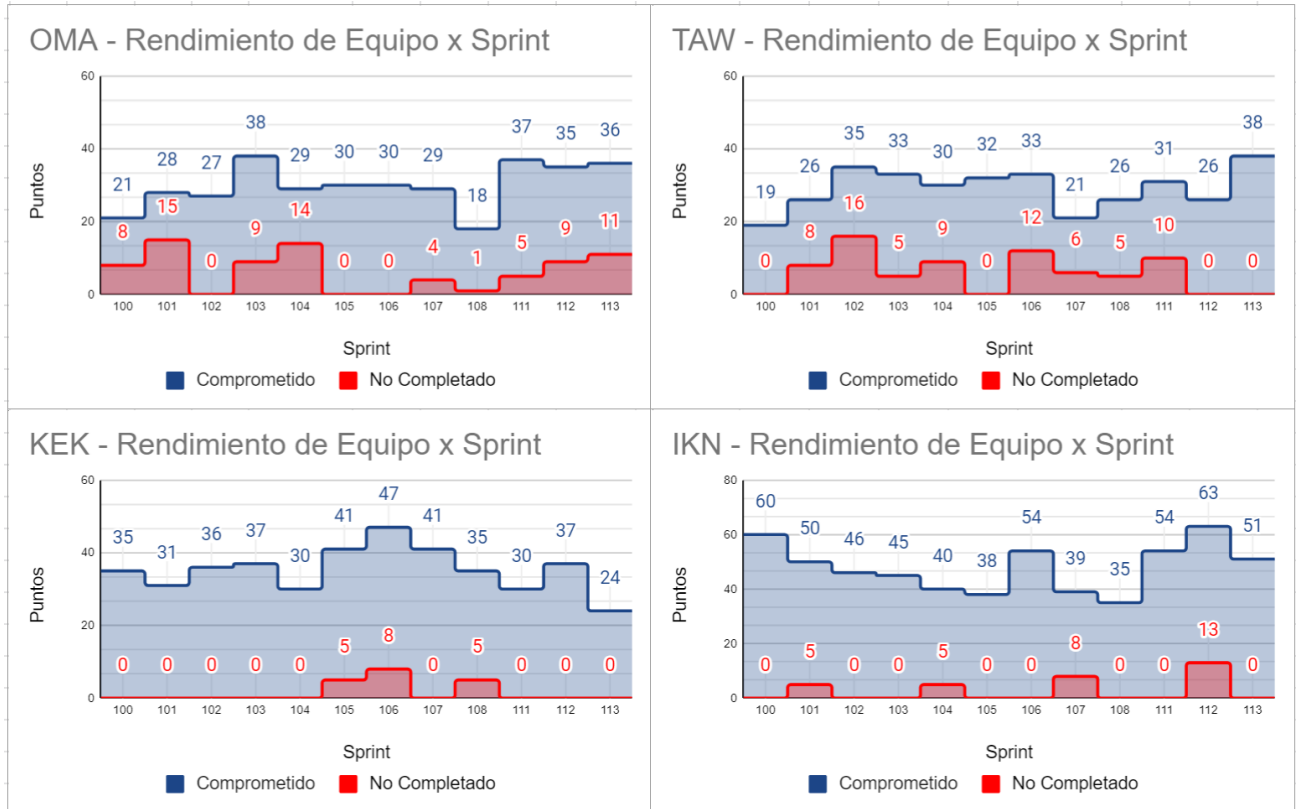
Tabla 1.8. Porcentaje de trabajo no completado por proyecto - Equipo IKN

Proyecto	Trabajo comprometido	Trabajo no completado	
Cliente Web Mobile	221	18	8%
Cliente Web CNB	121	10	8%
Cliente Web – Sec. Breach	113	3	3%
Retiro dinero sin tarjeta	21	8	38%
CSP CNB a Centralizado	53	0	0%

Nota: Adaptado de Reportes Jira.

En la Figura 1.6., se muestra un resumen de los resultados históricos de cada uno de los equipos. El gráfico de barras muestra en azul los puntos comprometidos y en rojo los puntos que no fueron completados en cada uno de los *sprints*.

Figura 1.6. Gráficos de rendimiento de equipos por sprint



Nota: Adaptado de Reportes Jira.

Es posible observar una relación proporcional entre el trabajo comprometido y el trabajo no completado en el *sprint* 101 de OMA, los *sprints* 102, 106 y 111 de TAW, el *sprint* 106 de KEK y el *sprint* 112 de IKN. De forma contraria, se observa que existen *sprints* de gran puntaje comprometido que se lograron completar en su totalidad, como lo son el *sprint* 113 de TAW, el *sprint* 107 de KEK y los *sprints* 100, 106 y 111 de IKN. Por lo tanto, es posible concluir que no existe una tendencia definida de comportamiento entre ambos indicadores.

PMI (2016) define que un requisito es una condición o capacidad que se requiere esté presente en el producto, servicio o resultado para satisfacer un contrato u otra especificación formal. Según Schwaber, K. y Sutherland, J. (2020) todo requisito que se trabaja en un *sprint* debe ser potencialmente entregable al usuario al final de este.

Cuando un requisito es incorporado al *sprint* por el equipo *Scrum*, se crea una expectativa y compromiso de entrega con el *Product Owner*, patrocinador y demás interesados del proyecto, quienes esperan la entrega correspondiente a la iteración o al menos observar una demostración de funcionamiento al finalizar el *sprint*, para brindar su retroalimentación al producto. Por lo tanto, es posible concluir que los porcentajes de trabajo no completado provocaron retrasos en las entregas iterativas de los proyectos e incumplimiento del compromiso de entrega a los interesados.

1.2.2 Enunciado del problema

Basado en los resultados del estudio, fue posible identificar que el problema que afecta a la Tribu Navi consiste en el incumplimiento en la entrega de los requisitos durante el *sprint*. Este incumplimiento impacta directamente al *Product Owner*, quién se ve afectado por no contar con los entregables en el *sprint* acordado, se ve obligado a incluir el trabajo no completado en los siguientes *sprints* y se ve imposibilitado de tomar decisiones correctivas con respecto al producto.

Afecta también al patrocinador del proyecto, quién tiene sus expectativas puestas en los tiempos de entrega de los proyectos, ya que ha adquirido compromisos con los clientes internos y externos. Por último, la problemática afecta al equipo *Scrum*, ya que sus indicadores de desempeño se ven disminuidos por no ser capaces de completar todos los requisitos comprometidos, tal y como se mostró en las Tabla 1.1, Tabla 1.3, Tabla 1.5 y Tabla 1.7, en las cuales los porcentajes de incumplimiento estuvieron entre el 4% y el 21%.

1.2.3 Efectos del problema

Es posible identificar tres efectos principales del problema:

- El primer efecto identificado consiste en el retraso provocado en los proyectos debido al incumplimiento de las entregas iterativas. En el análisis del porcentaje de trabajo no completado por proyecto se evidencia que un 70% de los proyectos presentan retrasos en sus entregas por *sprint*.
- El segundo efecto del problema consiste en el incumplimiento de los compromisos de entrega, ya que, en el 50% de las ocasiones en que los equipos *Scrum* comprometen una entrega del *sprint*, no logran completarla.
- Por último, se presenta un efecto del problema que disminuye el indicador de desempeño de los equipos *Scrum* entre un 4% y un 21%.

1.2.4 Análisis de causas del problema

Con el objetivo de determinar las causas del problema identificado, se realiza un análisis a detalle de cada uno de los requisitos no completados. Para esto, los *Scrum Master* se basa en la documentación de la resolución del requisito en Jira y en su propia documentación de impedimentos para determinar las causas que se presentan con mayor frecuencia.

En el Cuadro 1.2, se muestra un listado de las causas encontradas y su nivel de frecuencia. Se utilizan tres niveles de frecuencia para el análisis: siempre (más del 70% de los casos), regularmente (entre el 30% y el 70% de los casos) y en ocasiones (menos del 30% de los casos).

Según el Cuadro 1.2, existen tres causas más frecuentes que provocan que el trabajo comprometido no sea completado:

- La causa A, hace referencia a la complejidad de los requisitos. Cuando el equipo *Scrum* se encuentra trabajando durante el *sprint*, descubre que los requisitos son más complejos de

lo estimado y que requerirán de un esfuerzo que sobrepasa su capacidad para el *sprint* en curso. Esto evidencia que durante el *Sprint Planning* el equipo *Scrum* no logra identificar correctamente el nivel de complejidad del requisito.

- La causa B, hace referencia a la ambigüedad de los requisitos. Cuando el equipo *Scrum* estima el esfuerzo necesario para completar un requisito, se basa en la descripción brindada por el *Product Owner*. Si dicha descripción no es lo suficientemente clara, el equipo *Scrum* podría estimar incorrectamente el requisito o realizar un entregable que no satisface lo esperado por el usuario.
- Por último, la causa C, consiste en la dependencia de personas externas al equipo *Scrum*, la cual impide completar el requisito antes de la finalización del *sprint*. Esto evidencia que durante el *Sprint Planning* el equipo *Scrum* no logra identificar las dependencias de los requisitos o no coordina adecuadamente con las personas externas.

Cuadro 1.2. Causas del trabajo no completado

Identificador	Causa	Frecuencia
A	Los requisitos eran más complejos de lo previsto.	Regularmente
B	Los requisitos no eran lo suficientemente claros.	Regularmente
C	Existió dependencia de personas externas al equipo.	Regularmente
D	El equipo <i>Scrum</i> no contó con el compromiso, el enfoque, la apertura, el coraje y el respeto necesarios.	En ocasiones
E	El equipo <i>Scrum</i> no contó con la capacidad técnica necesaria.	En ocasiones
F	Existió dependencia entre los requisitos.	En ocasiones
G	Los recursos tecnológicos no estuvieron disponibles o no fueron eficientes.	En ocasiones
H	Se presentaron cambios de prioridad durante el Sprint	En ocasiones
I	Se presentaron cambios no contemplados en los requisitos.	En ocasiones
J	Existieron interrupciones, vacaciones o incapacidades no planificadas.	En ocasiones

Nota: Adaptado de Reportes Jira y la documentación de los Scrum Master.

En la Tabla 1.9 se muestran ejemplos de casos específicos que evidencian las causas del trabajo no completado que afectan los proyectos. Estos casos son recolectados de los reportes de Jira y la documentación de impedimentos de los *Scrum Master* y tienen como objetivo ejemplificar las causas más frecuentes del problema (A, B y C).

Tabla 1.9. Casos específicos que evidencian las causas del trabajo no completado

Equipo	Proyecto afectado	Trabajo no completado	Causa
OMA	SafeKey AX	8	C: Aprobación técnica no llegó a tiempo.
OMA	SafeKey AX	7	C: Visto bueno de DBA no llegó a tiempo.
OMA	Segregación	2	A: Desarrollo resultó ser más complejo.
OMA	Migración WAS 8.5	5	B: El alcance no estuvo claro.
TAW	Migración Automática	8	A: Hubo trabajo adicional no contemplado.
TAW	Migración Automática	11	A: Las pruebas fueron más complejas.
TAW	AVP SILP Centralizado	3	C: La aprobación del cambio no llegó a tiempo.
KEK	CSP - PIN ONLINE	8	A: No hubo tiempo suficiente para finalizar.
KEK	Pases AIS CSP	5	C: La aprobación del cambio no llegó a tiempo.
KEK	Check in - Check Out PAN	5	B: El requisito no estaba completo.
IKN	Cliente Web CNB	8	A: No hubo tiempo suficiente para finalizar.
IKN	Cliente Web Mobile	5	A: No hubo tiempo suficiente para finalizar.
IKN	Cliente Web – Sec. Breach	3	B: El requisito no estaba completo.

Nota: Adaptado de Reportes Jira y la documentación de los *Scrum Master*.

Adicionalmente, se determinan siete causas que se presentan con menor frecuencia, pero que también afectan directamente el trabajo realizado por los equipos *Scrum*:

- La causa D, hace referencia a los valores de *Scrum*, los cuales deben ser adoptados por el equipo en todo momento. La apreciación subjetiva de estos valores es realizada por el *Scrum Master*, quién, mediante una observación constante del comportamiento es capaz de detectar la ausencia de estos valores. Algunos ejemplos de esta ausencia de valores son la

baja proactividad del equipo para indagar los requisitos y la falta de acción al encontrarse frente a un impedimento durante el *sprint*.

- La causa E, implica que el equipo *Scrum* no cuenta con todas las capacidades técnicas necesarias para desarrollar los requisitos.
- La causa F, se refiere a que, si un requisito debe ser completado para iniciar otro, el equipo *Scrum* se verá expuesto al impacto de las dependencias entre dichos requisitos.
- La causa G, se refiere a que los recursos tecnológicos con los que el equipo *Scrum* cuenta no están disponibles para su utilización o presentan complicaciones que los hacen ineficientes. Ejemplos de estos impedimentos son la no disponibilidad de los ambientes de pruebas y la afectación del trabajo diario por problemas técnicos en los computadores.
- Las causas H e I, hacen referencia a la priorización de proyectos y al control de cambios. Idealmente, el equipo *Scrum* trabaja en un único proyecto por *sprint* para mantener el enfoque y los requisitos se mantienen sin cambios durante el *sprint*. A pesar de esto, el *Product Owner* tiene la potestad de solicitar un cambio de prioridad y cambios en los requisitos durante el *sprint* cuando así lo considere necesario. Si el cambio solicitado no se gestiona correctamente, el equipo se verá expuesto a incumplir con lo comprometido.
- Por último, la causa J, se refiere a la disponibilidad del equipo. Cuando el equipo *Scrum* se compromete a ejecutar una lista de requisitos considera su disponibilidad durante el *sprint*. Ahora bien, si durante el *sprint* se presentan interrupciones, vacaciones o incapacidades no contempladas originalmente, el equipo se expone a incumplir con lo comprometido.

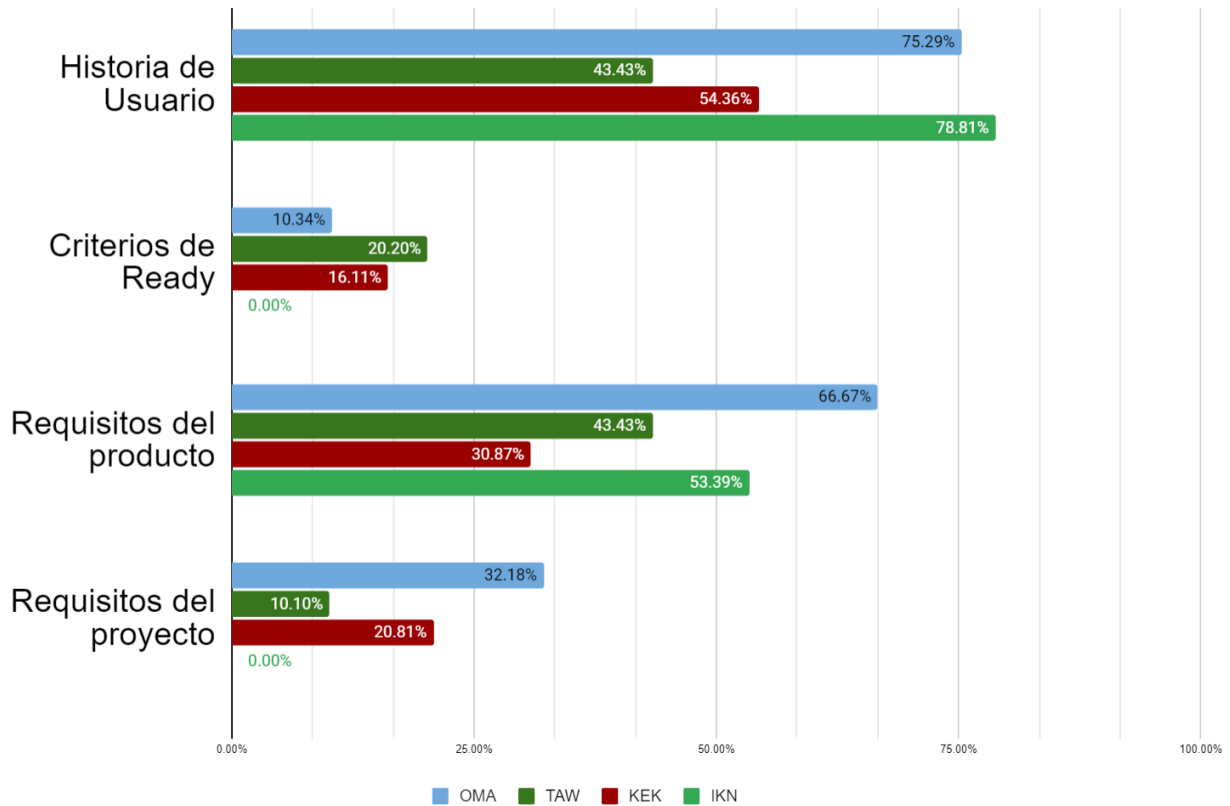
1.2.5 Análisis de causa raíz

Con el fin de identificar la causa raíz que provoca las causas descritas en el Cuadro 1.2, se realiza un estudio general del Manual Operativo *Scrum* de la organización para identificar las prácticas utilizadas y cómo los equipos *Scrum* las aplican.

Según el Manual Operativo *Scrum*, el evento denominado *Sprint Planning*, tiene como objetivo establecer un plan detallado del trabajo a realizarse durante el *sprint*. Durante este evento, el equipo se reúne para coleccionar los requisitos, definir el alcance del *sprint* y crear la estructura de desglose de trabajo. A pesar de que el Manual Operativo *Scrum* describe el *Sprint Planning*, no incluye procesos, técnicas ni herramientas que sirvan como insumo a los equipos *Scrum* para identificar, estimar, documentar y controlar los cambios de los requisitos, dejando al criterio de cada uno de ellos los métodos a utilizar.

Además, el Manual Operativo *Scrum* establece que todos los requisitos deben ser documentados en el software Jira, pero no define qué elementos debe contener la documentación ni brinda plantillas que ejemplifiquen cómo realizarlo correctamente. Con el fin de identificar cuales elementos incluyeron los equipos *Scrum* en la documentación, se realiza un análisis de los 540 requisitos que componen los 12 *sprints* del estudio. Los resultados de la Figura 1.7 evidencian que la documentación no se realiza de forma estandarizada para todos los requisitos y que cada equipo *Scrum* difiere en su concepto de documentación. El análisis evidencia también, que ninguno de los requisitos incluye documentación relacionada al control de cambios.

Figura 1.7. Documentación de requisitos



Nota: Adaptado de *Reportes Jira*.

PMI (2017) establece que un plan de gestión de requisitos es un componente del plan de gestión del proyecto que describe cómo serán analizados, gestionados y documentados los requisitos. El estudio de causa raíz realizado evidencia que los equipos *Scrum* no cuentan con un plan de gestión de requisitos para sus *sprints* debido a que el Manual Operativo *Scrum* no les brinda lo necesario para realizarlo.

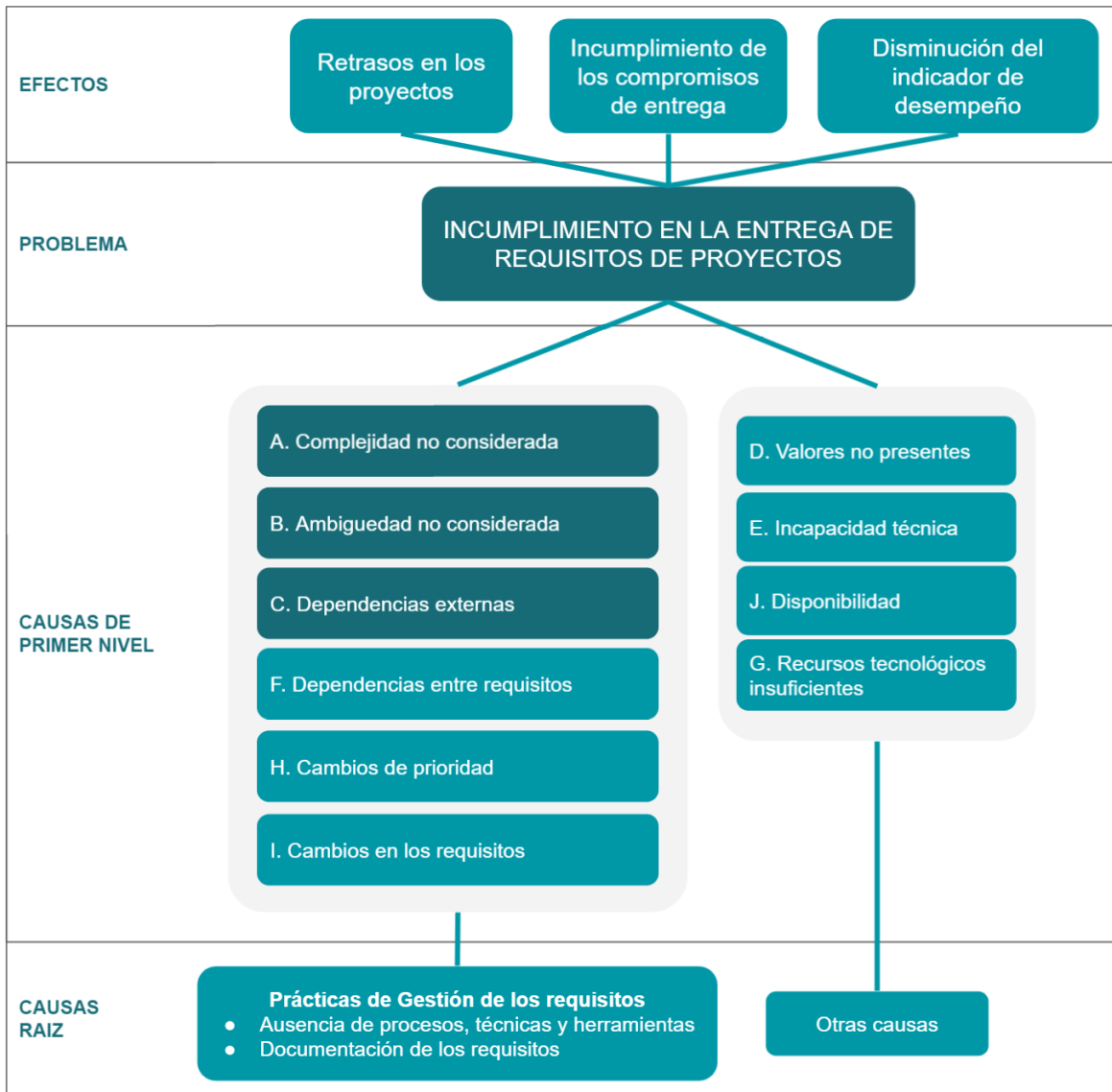
En conclusión, la causa raíz del problema radica en las prácticas actuales de gestión de los requisitos, las cuales no permiten realizar una correcta identificación, estimación, documentación y control de los requisitos. Esto provoca que los equipos *Scrum* planifiquen la entrega de requisitos con altos niveles de complejidad, ambigüedad y dependencias que provocan que el trabajo no sea completado a tiempo.

1.2.6 Árbol de problemas

Como se muestra en la

Figura 1.8. Árbol de **Problemas**, la problemática central radica en el incumplimiento en la entrega de los requisitos por *sprint* debido a causas relacionadas con la gestión de los requisitos y produce efectos negativos como retrasos en los proyectos, incumplimiento de los compromisos de entrega y disminución del indicador de desempeño de los equipos *Scrum*. Además, se muestra que existen otras causas raíz de menor impacto en la problemática, las cuales estarán fuera del alcance de la propuesta de solución.

Figura 1.8. Árbol de Problemas



1.3 Justificación del estudio

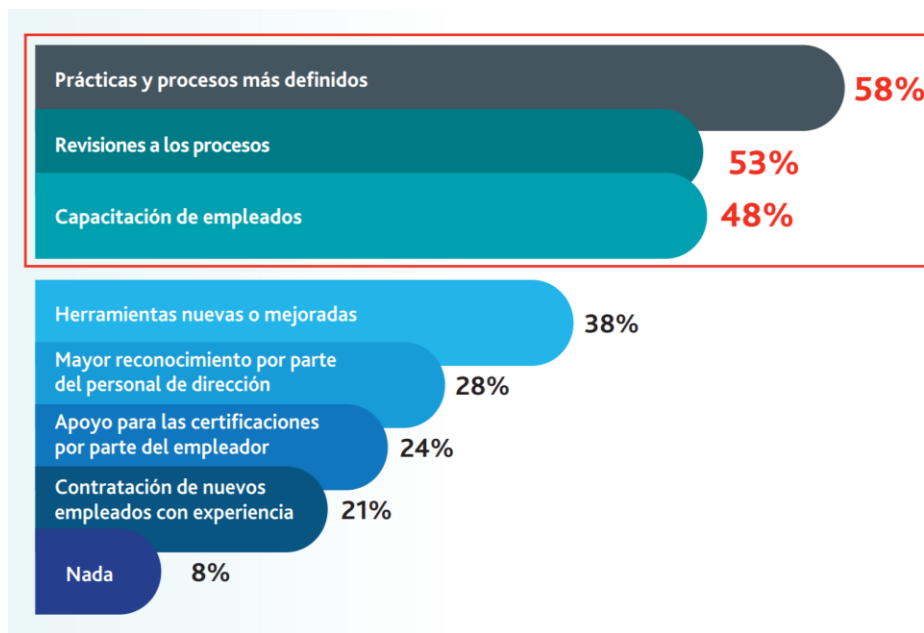
PMI (2017) menciona que “a medida que el entorno mundial se vuelve más complejo, las organizaciones están empezando a reconocer cómo utilizar el análisis de negocios para obtener ventajas competitivas mediante la definición, gestión y control de las actividades de los **requisitos.**” Además, establece que “los métodos ágiles deliberadamente invierten menos tiempo tratando de definir y acordar el alcance en la etapa temprana del proyecto e invierten más tiempo estableciendo el proceso para su descubrimiento y perfeccionamiento constantes.”

Con base en lo anterior, es posible afirmar que los equipos *Scrum* hacen bien al intentar definir sus requisitos de forma iterativa durante cada *sprint*, pero al no realizarlo de forma metódica y estructurada se exponen constantemente al incumplimiento de las entregas. PMI (2017) también establece que el éxito del proyecto depende “directamente de la participación de los interesados en el descubrimiento y la descomposición de las necesidades en requisitos del proyecto y del producto, y del cuidado que se tenga al determinar, documentar y gestionar los requisitos del producto.” Es por esto, que la solución a la problemática identificada es necesaria para poder alcanzar el éxito de los proyectos en la Tribu Navi.

IAG Consulting (2009) determina que un 74% de las empresas se clasifican con un nivel de madurez bajo en la gestión de requisitos, lo cual resulta en incumplimiento de tiempos de entrega, excesos en costos y desperdicio de recursos que impactan los proyectos. También, especifica que solo un 54% de las veces logran alcanzar sus objetivos y que les toma un 35% más de tiempo lograrlo. Es posible relacionar estos datos con la situación actual de los equipos *Scrum*, quienes sufren una disminución de su rendimiento debido al incumplimiento en las entregas y ven afectados sus proyectos por no ser capaces de entregar lo comprometido en cada *sprint*.

Según PMI (2014), la causa principal (37%) del fracaso de los proyectos en 2014 es la recolección inadecuada de requisitos. Este estudio establece que por cada dólar que se gasta en proyectos, el 5.1% se desperdicia debido a la gestión deficiente de los requisitos. Además, puntualiza que cuando los proyectos no cumplen con sus objetivos se debe a la gestión deficiente de los requisitos en un 47% de las ocasiones. Si no se presta atención a la problemática de la gestión de requisitos sufrida por la Tribu Navi y no se implementa una solución oportuna, es probable que se convierta en una de las principales causas del fracaso de los proyectos y genere desperdicios económicos.

Figura 1.9. Esfuerzos para la mejora de la gestión de los requisitos



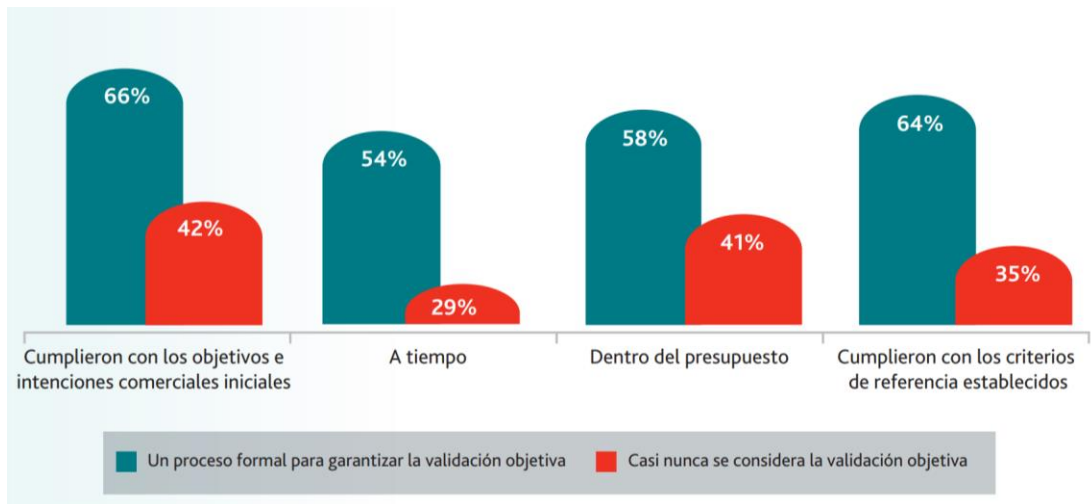
Nota: Tomado de *Pulso de la profesión*® 2014.

El estudio establece también que “las organizaciones deben estandarizar y formalizar sus procesos a nivel de proyectos y programas, para garantizar que apliquen constantemente buenas prácticas de gestión de requisitos en todas sus iniciativas.” Además, como se muestra en la Figura 1.9, halla que un 58% de las organizaciones se enfocan en definir mejores prácticas y procesos y un 53% en la revisión de los procesos actuales en busca de optimizar su desempeño en la gestión de requisitos. Cabe destacar que el estudio también identifica que las organizaciones hacen esfuerzos importantes en capacitar a sus empleados para que contaran con las competencias necesarias.

Como se muestra en la

Figura 1.10, el estudio también demuestra que aquellas organizaciones que aplican un proceso formal y prácticas de gestión para la validación de los requisitos logran resultados considerablemente mejores de los proyectos, incluyendo el cumplimiento de los objetivos, los tiempos de entrega, el presupuesto y los criterios de referencia.

Figura 1.10. Comparativa de resultados organizacionales con y sin procesos formales



Nota: Tomado de *Pulso de la profesión*® 2014.

El estudio, que incluye a 2066 directores de proyectos y programas, deduce que las organizaciones pueden mitigar el perjuicio que la deficiente gestión de requisitos tiene en sus proyectos y su consecuente despilfarro financiero, mediante la estandarización, formalización y mejora de los procesos de gestión de requisitos, aplicando constantemente buenas prácticas de gestión de requisitos. Con base en lo anterior, es posible afirmar que, para lograr mejores resultados en sus proyectos, la Tribu Navi debe realizar esfuerzos por implementar procesos y prácticas que aporten a la disminución de la problemática.

Con base en lo establecido por PMI (2017) y en los estudios mencionados, la solución a las causas de la problemática brindaría beneficios de consideración a la Tribu Navi, los cuales serán expuestos a continuación.

Como se muestra en la Figura 1.11, al implementar una solución a la problemática en la Tribu Navi se estima que será posible mitigar seis de las diez causas encontradas del trabajo no completado. Esto equivaldría a aproximadamente un 76% si se contempla la proporcionalidad de la frecuencia con la que se presenta cada causa expuesta en el Cuadro 1.2.

Figura 1.11. Causas mitigadas con la solución.



Nota: Adaptado de **Cuadro 1.2**

Por lo tanto, si la solución es implementada en la Tribu Navi y adoptada por los equipos *Scrum*, es posible estimar un aumento de las entregas realizadas a tiempo debido a una adecuada gestión de los requisitos. En la Tabla 1.10, se muestra una proyección del aumento estimado en las entregas a tiempo de los proyectos para el equipo OMA, asumiendo una disminución del 76% en el trabajo no completado.

Tabla 1.10. Proyección de las entregas de proyectos a tiempo – Equipo OMA

Proyecto	Entregas a tiempo sin la solución	Entregas a tiempo con la solución
SafeKey AX	82%	95.68%
Tarjetas Clave	83%	95.92%
Segregación	82%	95.68%
Migración WAS 8.5	74%	93.76%
3DS 2.0	42%	86.08%

Nota: Adaptado de *Reportes Jira*.

A partir de esta proyección es posible deducir que, la solución trae consigo mejoras en el desempeño de los proyectos, gracias al aumento de las entregas realizadas a tiempo, así como, el aumento de la satisfacción de las expectativas del *Product Owner*, patrocinador y demás interesados. Además, permite una mejora del indicador de desempeño de los equipos *Scrum*, los cuales están capacitados para alcanzar resultados superiores al 95% de trabajo completado por *sprint* en promedio.

Finalmente, la solución permite la estandarización de los procesos de gestión de los requisitos en la Tribu Navi. Esto le habilita la posibilidad a futuro, de medir, controlar y mejorar sus procesos. Además, fortalece los activos de proyectos de la organización, ya que, al contar con insumos para la gestión estandarizada de los requisitos de proyectos, se pueden beneficiar equipos *Scrum* de otras tribus.

De no implementarse una solución a la problemática expuesta de manera oportuna, la Tribu Navi se expone a mantener la tendencia negativa en el incumplimiento de los plazos de entrega de

los proyectos, la insatisfacción de los interesados y la degradación del desempeño de sus equipos *Scrum*.

1.4 Objetivos

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

1.4.1 Objetivo general

Proponer un método estándar de gestión de requisitos de proyectos en la Tribu Navi mediante el rediseño de sus procesos, técnicas y herramientas para la disminución del porcentaje de incumplimiento de las entregas por *sprint*.

1.4.2 Objetivos específicos

1. Identificar las prácticas de gestión de requisitos de proyectos utilizadas actualmente por los equipos *Scrum* mediante el análisis de los 12 *sprints* del estudio para la descripción de sus procesos, técnicas, herramientas y elementos que influyen en la problemática.
2. Exponer las buenas prácticas de gestión de requisitos de proyectos acordes al marco de trabajo *Scrum* mediante la revisión bibliográfica para ser incorporadas en los procesos, técnicas y herramientas actuales.
3. Rediseñar los procesos, técnicas y herramientas de gestión de requisitos de proyectos por medio de la incorporación de las buenas prácticas expuestas para la generación de un método estándar.
4. Desarrollar una estrategia de implementación del método de gestión de requisitos de proyectos mediante la descripción de las actividades, presupuesto y responsables para la ejecución de la propuesta en la Tribu Navi.

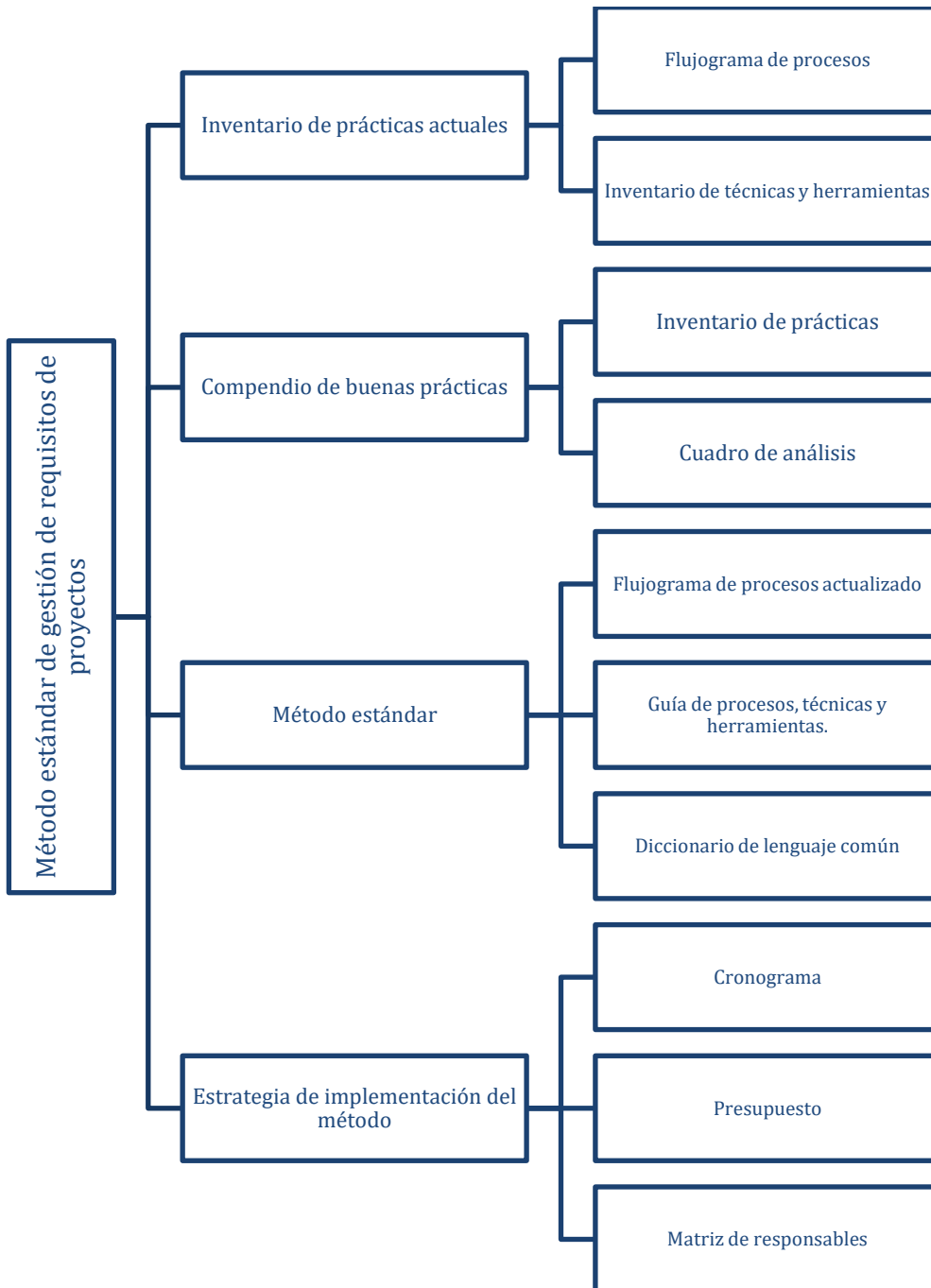
1.5 Alcance y limitaciones

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

1.5.1 Alcance

El presente proyecto tiene como entregable general un método estándar de gestión de requisitos de proyectos para uso de la Tribu Navi. El método servirá como insumo a los equipos *Scrum* para gestionar de forma ordenada y sistemática los requisitos de proyectos trabajados en cada *sprint*. Además, el método pretende estandarizar las prácticas de gestión de requisitos de proyectos ejecutadas por los equipos *Scrum*. En la Figura 1.12, se muestra la estructura de desglose de los entregables que forman parte del alcance.

Figura 1.12. Estructura de desglose de los entregables



El inventario de prácticas actuales pretende identificar las prácticas de gestión de requisitos de proyectos adoptadas por la Tribu Navi durante los *sprints* en los que es identificada la problemática. Se compone de dos sub-entregables: el flujograma de procesos y el inventario de técnicas y herramientas. El flujograma de procesos consiste en una representación gráfica de la secuencia de las actividades de los procesos de gestión de requisitos de proyectos. Por su parte, el inventario de técnicas y herramientas consiste en un listado ordenado de las técnicas y herramientas ejecutadas por la Tribu Navi para la gestión de requisitos de proyectos. Estos listados incluyen el nombre de la técnica o herramienta, el proceso y evento *Scrum* en el cual se utiliza, una descripción de la técnica o herramienta y un análisis de los elementos que influyen en la problemática.

El compendio de buenas prácticas se compone de dos sub-entregables. El primero consiste en una lista de prácticas reconocidas para la correcta gestión de los requisitos de proyectos, así como una explicación breve pero precisa de su funcionamiento. El segundo es un cuadro de análisis, el cual incluye los elementos de la problemática que serán resueltos con cada una de las buenas prácticas.

El método estándar se compone de tres sub-entregables: el flujograma de procesos actualizado, la guía de procesos, técnicas y herramientas; y el diccionario de lenguaje común. El flujograma de procesos actualizado representa de forma gráfica la nueva secuencia de actividades de los procesos de gestión de requisitos de proyectos. La guía incluye una explicación de los nuevos procesos, técnicas y herramientas; y cómo estos se adecuan al marco de trabajo *Scrum*. Por último, el diccionario consiste en un listado de conceptos relacionados a la gestión de requisitos de proyectos que tiene como objetivo facilitar la comprensión de la guía mediante el uso de un lenguaje común en la Tribu Navi.

La estrategia de implementación está compuesta por una descripción de las actividades necesarias para poner en marcha el nuevo método de gestión de requisitos de proyectos en la Tribu Navi, así como del cronograma y el presupuesto correspondiente.

Están fuera del alcance, la implementación del método en la Tribu Navi, la capacitación del personal para desarrollar las nuevas competencias requeridas, la creación de indicadores y la estrategia futura de mejora de los procesos.

1.5.2 Limitaciones

El método debe ser adecuado a los principios ágiles de desarrollo de software y al marco de trabajo *Scrum*, ya que la organización definió hace más de cuatro años que esta es la metodología estándar para desarrollo de soluciones tecnológicas.

La Tribu Navi cuenta con *Scrum Master* encargados de dirigir las iniciativas de mejora de la metodología de proyectos, por lo cual, el método debe ser implementado bajo su dirección y no será válida la contratación de personal externo para esta función.

El método no incluye la creación de nuevos puestos de trabajo, pues los cambios en la estructura organizacional de la Tribu Navi son aprobados por los directivos y debido a la situación de la pandemia, se han congelado las contrataciones de nuevo personal durante el 2021.

El método contempla la utilización de las herramientas tecnológicas de gestión de proyectos actuales, ya que son de uso común para todos los equipos *Scrum* de la organización y existen contratos vigentes con los proveedores.

Capítulo 2 Marco teórico

En este capítulo se exponen los conceptos teóricos que fundamentan la presente investigación, incluyendo los temas relacionados a la problemática y a la solución propuesta.

Primeramente, se brinda una visión general del marco de trabajo *Scrum* como guía metodológica y se exponen a detalle sus pilares, roles, eventos y artefactos. Posteriormente, se presentan los conceptos relacionados a la gestión de requisitos, sus procesos y relaciones con la gestión del proyecto.

Una vez comprendidos los conceptos de la gestión de requisitos, se indican las generalidades, procesos y buenas prácticas de la especificación de requisitos de sistemas desde una perspectiva general. Posteriormente, se detallan los componentes y consideraciones a tomar en cuenta durante la formulación de especificaciones de requisitos de software.

Finalmente, se brinda un vistazo general a los conceptos relacionados al software de gestión Jira y sus componentes, con el objetivo de comprender sus funcionalidades relacionadas a la gestión de requisitos.

2.1 Marco de trabajo Scrum

Las metodologías ágiles de gestión de proyectos son adoptadas por las organizaciones en busca de mejorar su capacidad de adaptación a los cambios constantes del entorno y mantenerse vigentes en el mercado. Según PMI (2017), los principios ágiles definen como su prioridad la satisfacción del cliente y esto permite generar productos y servicios que lo deleitan. Además, la retroalimentación cíclica, rápida y transparente permite a la organización enfocarse en la experiencia del cliente y así mantenerse competitiva y relevante.

El marco de trabajo *Scrum* es utilizado mundialmente por organizaciones de desarrollo de software que buscan establecer una metodología ágil, la cual les permita satisfacer las necesidades de negocio de una manera eficaz. Según Schwaber, K. y Sutherland, J. (2020) *Scrum* es un marco de trabajo diseñado para resolver problemas complejos mediante soluciones adaptativas, basado en el empirismo y en el pensamiento *Lean*.

2.1.1 Pilares

Según Schwaber, K. y Sutherland, J. (2020) *Scrum* cuenta con tres pilares que deben ser comprendidos y respetados para alcanzar el éxito en su implementación. Los tres pilares son:

- **Transparencia:** El proceso y el trabajo emergentes deben ser visibles, ya que, aquellos artefactos de *Scrum* que tienen poca transparencia pueden llevar a decisiones que disminuyan el valor y aumenten el riesgo. La transparencia permite la inspección.
- **Inspección:** Los artefactos de *Scrum* y el progreso hacia los objetivos acordados deben examinarse con frecuencia y diligencia para detectar variaciones o problemas. La inspección permite la adaptación.
- **Adaptación:** Si algún aspecto de un proceso se desvía fuera de los límites aceptables o si el producto resultante es inaceptable, el proceso que se aplica o los materiales que se producen deben ajustarse. Se espera que un equipo *Scrum* se adapte en el momento en que aprenda algo nuevo a través de la inspección.

2.1.2 Valores

Según Schwaber, K. y Sutherland, J. (2020) existen cinco valores que deben ser adoptados por los miembros del equipo *Scrum* en busca de adecuar su cultura individual a la cultura del

equipo, con el fin de alcanzar sus objetivos planteados. Estos valores son el compromiso, el enfoque, la franqueza, el respeto y el coraje.

2.1.3 Roles

Schwaber, K. y Sutherland, J. (2020) establecen tres roles de equipo que deben estar siempre presentes y definen un conjunto de responsabilidades para cada uno de los miembros del equipo. Los roles de equipo son:

- *Developers*: Son los miembros del equipo que se comprometen a crear cualquier aspecto de un incremento utilizable en cada *sprint*. Son responsables de crear un plan para el *sprint*, conocido como *sprint backlog*, inculcar calidad al adherirse a una definición de terminado, adaptar su plan cada día hacia el objetivo del *sprint* y responsabilizarse mutuamente como profesionales.
- *Product Owner*: Es responsable de maximizar el valor del producto resultante del trabajo del equipo *Scrum*, de desarrollar y comunicar explícitamente el objetivo del producto, de crear y comunicar claramente los elementos del *product backlog*, de ordenar los elementos del *product backlog* y de asegurarse de que el *product backlog* sea transparente, visible y se entienda.
- *Scrum Master*: Es responsable de implementar *Scrum* como se define en la Guía de *Scrum*, de lograr la efectividad del equipo *Scrum*, de guiar a los miembros del equipo en ser autogestionados y multifuncionales; de ayudar al equipo *Scrum* a enfocarse en crear incrementos de alto valor que cumplan con la definición de terminado; de procurar la eliminación de impedimentos y de asegurarse de que todos los eventos de *Scrum* se lleven a cabo y sean positivos, productivos y se mantengan dentro de los límites de tiempo.

2.1.4 Eventos

Schwaber, K. y Sutherland, J. (2020) definen cinco eventos que conforman el marco de trabajo, los cuales deben ser comprendidos y respetados por los miembros del equipo. Estos eventos son:

- *Sprint*: Este incidente contiene al resto de eventos, tiene una duración fija menor a un mes e inicia inmediatamente después de la conclusión del *sprint* anterior. Durante este, no se realizan cambios que pongan en peligro el objetivo, la calidad del producto no disminuye, el *product backlog* se refina según sea necesario y el alcance se puede aclarar y renegociar con el *Product Owner* a medida que se aprende más.
- *Sprint Planning*: Establece el trabajo que se realizará durante el Sprint. El equipo *Scrum* crea este plan mediante trabajo colaborativo. El *Product Owner* se asegura de que los asistentes estén preparados para discutir los elementos más importantes del *Product Backlog* y cómo se relacionan con el objetivo del producto. El equipo *Scrum* también puede invitar a otras personas a asistir al *sprint planning* para brindar asesoramiento. Durante evento se deberá contestar a las siguientes preguntas: ¿Por qué es valioso este *sprint*?, ¿Qué se puede hacer en este *sprint*? y ¿Cómo se realizará el trabajo elegido?
- *Daily Scrum*: El propósito es inspeccionar el progreso hacia el objetivo del *sprint* y adaptar el *sprint backlog*. Este evento produce un plan viable para el siguiente día de trabajo. PMI (2017) recomienda que algún miembro del equipo facilite el *daily Scrum* en lugar de un líder o gerente de proyectos para evitar que se transforme en una sesión de seguimiento del proyecto.
- *Sprint review*: El propósito del *sprint review* es inspeccionar el resultado del *sprint* y determinar futuras adaptaciones. El equipo *Scrum* presenta los resultados de su trabajo a los interesados clave y se discute el progreso hacia el objetivo del producto. Durante el

evento, el equipo *Scrum* y los interesados revisan lo que se logró en el *sprint* y lo que ha cambiado en su entorno. Con base en esta información, los asistentes colaboran sobre qué hacer a continuación. El *product backlog* también se puede ajustar para satisfacer nuevas oportunidades.

- *Sprint retrospective*: El propósito del *sprint retrospective* es planificar formas de aumentar la calidad y la efectividad. El equipo *Scrum* inspecciona cómo fue el último *sprint* con respecto a las personas, las interacciones, los procesos, las herramientas y su definición de terminado. PMI (2017) menciona que se podrían realizar retrospectivas adicionales cuando sea necesario, en especial cuando se realiza una entrega, cuando el trabajo no progrese debido a impedimentos o cuando se alcancen hitos importantes del proyecto.

2.1.5 Artefactos

Por último, Schwaber, K. y Sutherland, J. (2020) definen tres artefactos Scrum. Estos artefactos son:

- *Product backlog*: El *product backlog* es una lista emergente y ordenada de lo que se necesita para mejorar el producto. Es la única fuente del trabajo realizado por el equipo *Scrum*.
- *Sprint backlog*: El *sprint backlog* se compone del objetivo del *sprint* (¿por qué?), el conjunto de elementos del *product backlog* seleccionados para el *sprint* (¿qué?), así como un plan de acción para entregar el incremento (¿cómo?).
- Incremento: Un incremento es un peldaño concreto hacia el objetivo del producto. Cada incremento se suma a todos los incrementos anteriores y se verifica minuciosamente, en busca de garantizar que todos funcionen juntos. Para proporcionar valor, el incremento

debe ser utilizable por el usuario, por lo que el trabajo no puede considerarse parte de un incremento si no cumple con la definición de terminado.

2.2 Gestión de Requisitos

La guía *Requirements Management* del PMI tiene como propósito exponer los elementos que componen la gestión de requisitos, así como su criticidad para el éxito de proyectos y programas. Esta guía describe las actividades, conocimientos y habilidades necesarias para realizar una correcta gestión de requisitos, discute su importancia y su interrelación con la gestión de proyectos. Debido al reconocimiento mundial del PMI, esta guía fungirá como base para el desarrollo de los conceptos teóricos relacionados a la gestión de requisitos y serán incorporados conceptos de distintos autores para sustentar y ejemplificar cada subsección.

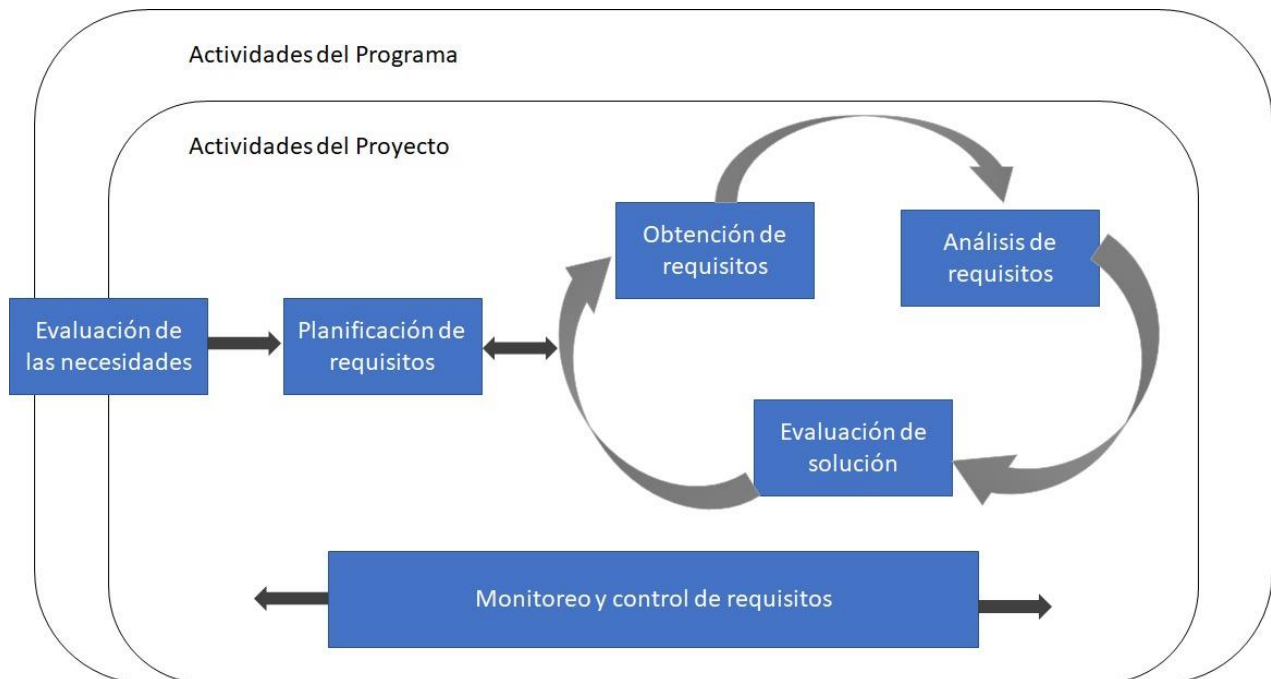
La guía define que un requisito es una condición o capacidad que se requiere esté presente en el producto, servicio o resultado para satisfacer un contrato u otra especificación formal. Además, establece que la gestión de requisitos consiste en todas aquellas actividades para desarrollar y gestionar los requisitos de un proyecto, que permiten establecer una línea base, trazabilidad y control de cambios de los requisitos. Young, R. (2004), establece que un requisito es un enunciado que identifica una capacidad, característica o una cualidad de un sistema que busca brindar valor y utilidad al cliente o usuario. Además, establece que un plan de gestión de requisitos define como evolucionan los requisitos reales y como se abordarán las actividades de requisitos.

2.2.1 Proceso de gestión de requisitos

PMI (2016) establece un proceso de gestión de requisitos, el cual está compuesto de un conjunto de actividades estandarizadas y estructuradas para desarrollar y gestionar los requisitos de un proyecto. El proceso parte de la premisa de que se conoce solo una parte de la información

de los requisitos inicialmente, por lo que establece iteraciones para descubrir más información acerca del producto, servicio o resultado y así adaptar los supuestos originales según el nuevo conocimiento adquirido. En la Figura 2.1 se observa el proceso de gestión de requisitos, su flujo iterativo y la relación con el proyecto y el programa.

Figura 2.1. Proceso de gestión de requisitos



Nota: Adaptado de la guía *Requirements Management* (2016).

Las actividades del proceso son ejecutadas de forma independiente o iterativa según lo dicte el proyecto. A continuación, se detallan estas actividades.

2.2.1.1 Evaluación de las necesidades

PMI (2016) establece que la evaluación de las necesidades consiste en la identificación y definición de un problema o una oportunidad de negocio. A pesar de que esta necesidad usualmente es descubierta de forma previa al inicio del proyecto,

es posible que factores externos provoquen cambios durante la ejecución del proyecto, por lo que las necesidades deben ser revisadas para asegurar que siguen vigentes. El objetivo de realizar una evaluación de necesidades consiste en obtener una definición de alto nivel de la necesidad que permita determinar una o varias alternativas de solución.

La evaluación de necesidades inicia con las actividades a nivel de portafolio, ya que se establecen los supuestos, restricciones, dependencias y prioridades que influyen en los requisitos. Por su parte, la hoja de ruta del portafolio debe proveer el contexto para los esfuerzos de gestión de requisitos del proyecto.

Posteriormente, se realizan las actividades a nivel de programa, en donde el caso de negocio provee una visión de las opciones de solución, tiempos esperados, limitaciones y filosofía detrás de la necesidad. La definición de los requisitos es mejorada gracias al entendimiento de las opciones de solución, el compromiso de los interesados y el registro de beneficios esperados.

2.2.1.2 Planificación de la gestión de requisitos

Según PMI (2016), la planificación de la gestión de requisitos tiene como objetivo brindar guía y dirección del cómo serán desarrollados y gestionados los requisitos durante el proyecto. El plan de gestión de requisitos es desarrollado, revisado y actualizado para establecer las actividades correspondientes a la obtención, análisis, monitoreo y control de los requisitos.

PMI (2017) establece que un plan de gestión de requisitos consiste en un componente del plan de gestión del proyecto en el cual se describe cómo serán analizados, documentados y gestionados los requisitos. Un plan de gestión de

requisitos exitoso requiere de una ejecución integrada con el plan de gestión del proyecto y del compromiso de los interesados.

Existen cuatro actividades que forman parte de la planificación:

- El análisis de los interesados es el proceso de analizar e identificar a las personas, grupos u organizaciones que puede afectar, ser afectadas, o tener algún interés en el proyecto. El plan de gestión de requisitos debe permitir entender las expectativas de los interesados clave para asegurar que las actividades planificadas sean las adecuadas para satisfacerlas.
- El inicio de la planificación consiste en obtener la información relacionada al proyecto e identificar los estándares y guías de la organización. Esta información puede estar conformada por los planes estratégicos de la organización, estudios de factibilidad, evaluación de las necesidades, casos de negocio y el plan de gestión del proyecto, entre otros.
- El desarrollo del plan de gestión de requisitos determina cómo los requisitos serán desarrollados, monitoreados, controlados, validados, reportados, priorizados, aprobados y mantenidos. Define el proceso de autorización y toma de decisiones, así como los roles y responsabilidades de los participantes. Además, establece como los requisitos serán documentados y comunicados a los interesados.
- El lanzamiento del plan de gestión de requisitos consiste en la revisión del plan con los interesados clave para obtener su aprobación en conjunto con el patrocinador. La aprobación del plan puede ser formal o informal y debe contar con un proceso definido.

2.2.1.3 Obtención de los requisitos

Según PMI (2016) la obtención de los requisitos consiste en el proceso de extracción de la información necesaria para desarrollar los requisitos de la solución, pretende identificar las condiciones y preferencias de la solución por parte de los interesados para así comprender la necesidad de negocio.

Este proceso se suele ejecutar de forma iterativa, ya que no existe certeza de los requisitos al inicio del proyecto. Conforme se descomponen los requisitos y se obtiene más información es posible que surjan nuevos requisitos. En un ciclo de vida adaptativo, la obtención y análisis de los requisitos se realiza durante la ejecución del proyecto como parte de la definición inicial del *Product backlog* y su refinamiento.

Existen cuatro actividades que forman parte de la obtención de los requisitos:

- La planificación para la obtención de requisitos consiste en definir el cómo se adquirirá la información necesaria para desarrollar los requisitos. El plan debe incluir las actividades a ejecutar, las fuentes de información desde las cuales se desarrollarán los requisitos, las personas que participarán y los entregables esperados.
- La clasificación de los requisitos pretende brindar mayor claridad en la definición de la información que se requiere obtener, así como determinar el nivel de detalle requerido para desarrollar la solución. Usualmente, se suelen clasificar en requisitos de negocio, requisitos de interesados, requisitos de la solución y requisitos de transición.
- La ejecución de las actividades de obtención pretende identificar aquella información implícita u oculta mediante la comunicación con los interesados, la revisión de sistemas, los registros históricos y la documentación.

- La documentación y comunicación de los resultados pretende asegurar su validez en conjunto con los interesados. Suelen utilizarse formularios, registros de audio, minutas de reuniones, notas de entrevistas y respuestas de encuestas para documentar la información obtenida.

Young, R. (2004) recomienda utilizar técnicas de obtención de requisitos como entrevistas, análisis de documentos, lluvia de ideas, talleres de requisitos, prototipos, casos de uso, análisis de interfaces, modelado, análisis de rendimiento y capacidad o escenarios.

2.2.1.4 Análisis de requisitos

Según PMI (2016) el análisis de requisitos consiste en la examinación, descomposición y síntesis de la información de los requisitos obtenida, en busca de alcanzar un conjunto accionable de requisitos que satisfaga las metas y objetivos planteados. El análisis debe realizarse de forma progresiva e iterativa en busca de alcanzar el nivel de detalle de los requisitos necesario para alcanzar la solución.

Existen tres actividades que forman parte del análisis de los requisitos:

- La planificación define la estrategia del análisis de requisitos, determina las actividades, técnicas y herramientas que serán utilizadas. Estas actividades deben ser refinadas y ajustadas conforme se descubren mayores detalles del proyecto.
- La ejecución de las actividades de análisis incluye seis componentes principales:
 - Identificación, análisis y documentación: Los atributos son características específicas que capturan información clave del requisito. Estos atributos ayudan a la trazabilidad y monitoreo durante el ciclo del proyecto.

- Selección del modelo: Los modelos son representaciones gráficas o textuales que permiten identificar información extraña o ausencia de información.
- Priorización: Los requisitos deben ser ordenados según su importancia en busca de facilitar el análisis del valor relativo de un requisito con respecto a otro. Esto permite a los interesados mantener el enfoque en los requisitos más críticos con base en los criterios de priorización. Usualmente, estos criterios son el valor, el nivel de riesgo, la complejidad, el costo y las restricciones regulatorias.
- Asignación y derivación: Los requisitos son asignados a entregas o iteraciones específicas. Esto asegura que la solución maximice el valor de negocio en cada entrega. La derivación consiste en descomponer los requisitos en otros más discretos de forma que sea disminuida la ambigüedad.
- Verificación: Los requisitos deben ser revisados para asegurar que han sido construidos correctamente y están libres de errores y omisiones. Un requisito bien construido no debe ser ambiguo, debe ser consistente, correcto, completo, medible, factible, trazable, preciso y comprobable. En un ciclo de vida adaptativo, un requisito redactado como historias de usuario debe ser independiente, negociable, de valor, estimable, pequeño y comprobable.

- Validación: Los requisitos deben ser validados para asegurar que reflejan las intenciones de los interesados y que permitirán satisfacer sus expectativas.
- La documentación de los requisitos es utilizada por el equipo de proyecto para desarrollar la solución. Esta documentación puede consistir en especificaciones, textos o modelos, ya sean formales o informales.

2.2.1.5 Monitoreo y control de requisitos

Según PMI (2016) los requisitos deberán ser continuamente rastreados, monitoreados y controlados de manera que se asegure que el proyecto satisface el alcance del producto, servicio o resultado y que los cambios en los requisitos son incluidos en el alcance cuando son aprobados.

La inestabilidad de los requisitos es una de las principales causas de variación del alcance del proyecto. PMI (2017) establece que a medida que aumenta la incertidumbre en el proyecto, también lo hacen la probabilidad de cambios, el trabajo desperdiciado y la reelaboración, lo que es costoso y requiere mucho tiempo.

El monitoreo y control de los requisitos facilita gestionar esta variación mediante el proceso de control integrado de cambios. En un ciclo de vida adaptativo, el alcance es controlado mediante la revisión del *Product backlog* en cada iteración, habilitando la posibilidad de que surjan nuevos requisitos.

PMI (2016) establece seis actividades que forman parte del monitoreo y control de los requisitos:

- La preparación del monitoreo y control consiste en definir los métodos y herramientas que serán ejecutados durante las etapas de obtención, análisis y evaluación para dar trazabilidad a los requisitos.
- La creación de la matriz de trazabilidad es realizada durante las etapas de obtención y análisis de los requisitos. Cuando un requisito es creado o actualizado, sus atributos son capturados en la matriz de trazabilidad.
- La aprobación de los atributos de requisitos recolectados y documentados por parte de los interesados es crítica para su incorporación en la línea base de los requisitos. La línea base permite al equipo del proyecto reconocer cuando ha ocurrido un cambio. En un ciclo de vida adaptativo, el *Product Owner* es el principal punto de contacto para aprobar los requisitos.
- La gestión de cambio consiste en el proceso que asegura que los cambios son gestionados y los requisitos son actualizados una vez que el cambio es aprobado. En un ciclo de vida adaptativo, el proceso de gestión de cambio se ejecuta mediante sesiones de planificación en las cuales se revisa el *Product backlog* y se prioriza el conjunto de requisitos a ser trabajados en el siguiente *sprint*.
- El monitoreo del estatus de los requisitos debe ejecutarse de manera periódica.
- La documentación y comunicación de los resultados mantiene informados a los interesados del estado actual y aumenta su nivel de colaboración. Deben documentarse y comunicarse toda actualización a la línea base de los requisitos y sus solicitudes de cambio.

2.2.1.6 Evaluación de la solución

Según PMI (2016), la evaluación de la solución se compone de las actividades desarrolladas para validar una solución que será o ha sido implementada. La evaluación determina que tan bien la solución satisface las necesidades de negocio expresadas por los interesados, por lo que podría resultar en la generación de nuevos requisitos o modificación de los existentes.

Existen tres actividades que forman parte de la evaluación de la solución:

- La planificación consiste en la selección de técnicas para la evaluación de la solución.
- La validación asegura que la solución funciona como se espera. En un ciclo de vida adaptativo, la validación se realiza al finalizar cada *sprint* cuando el equipo de proyecto brinda a los interesados una serie de requisitos listos para ser puestos en producción.
- La documentación de la evaluación depende de factores como el tamaño y la complejidad del proyecto, así como la naturaleza de la solución, restricciones regulatorias y el tipo de ciclo de vida.

2.2.1.7 Cierre del proyecto o fase

El cierre del proyecto o fase se realiza una vez el producto, servicio o resultado ha sido enviado a un estado operacional y las actividades de evaluación de la solución han sido ejecutadas para asegurar que la solución continúa satisfaciendo la necesidad de negocio y entregando el valor esperado.

El cierre es parte de la gestión del proyecto y por lo tanto responsabilidad del administrador del proyecto. El cierre debe incluir la documentación de lecciones

aprendidas para la transferencia de conocimiento, el soporte de la transición del producto a operaciones y la habilitación del mantenimiento de beneficios a largo plazo.

PMI (2017) establece que en los proyectos iterativos, adaptativos y ágiles el cierre prematuro de un proyecto no es un fracaso, ya que habrá generado un cierto valor útil, siempre y cuando hayan sido entregados aquellos requisitos de mayor valor para el negocio.

2.2.2 Requisitos y el ciclo de vida del proyecto

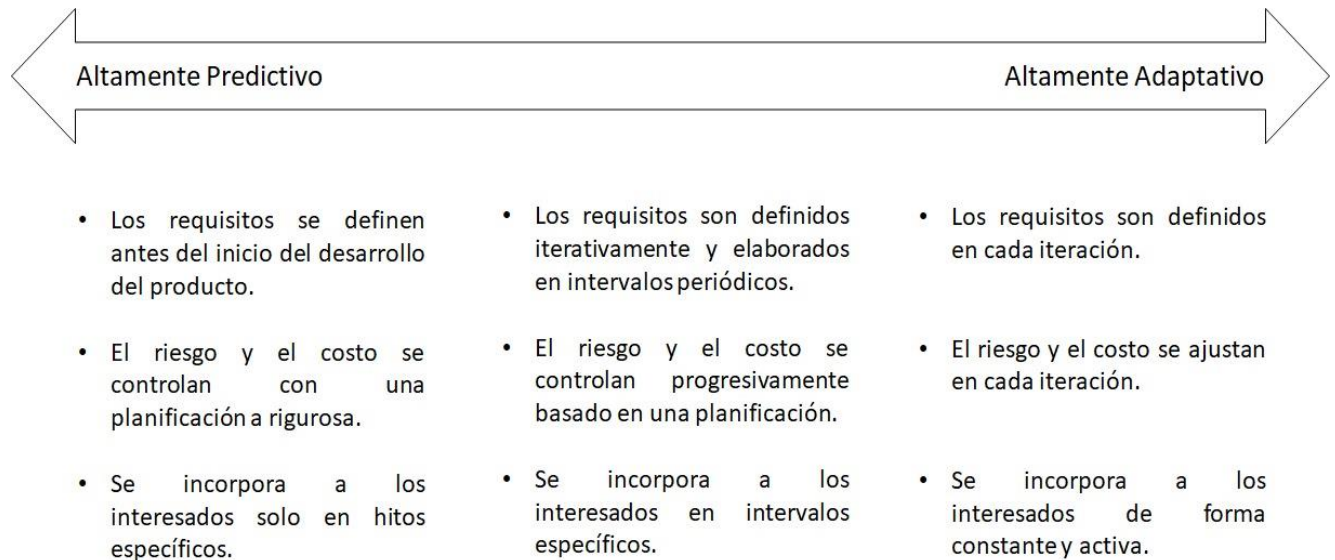
La gestión de requisitos debe estar alineada al ciclo de vida del proyecto y considerar sus características de complejidad y niveles de riesgo. El tipo de ciclo de vida del proyecto debe ser claramente definido ya que impactará la forma de gestionar los requisitos.

Si el ciclo de vida es predictivo, la obtención y análisis de los requisitos son completados previo al inicio del diseño y desarrollo de la solución, se requiere un único ciclo de análisis de requisitos y se utiliza un control de cambios formal.

Si el ciclo de vida es adaptativo, la obtención y análisis de requisitos se realiza durante el transcurso del proyecto, lo que provee mayor oportunidad de identificar las necesidades de negocio y comprender la solución requerida. Este tipo de ciclo de vida implica un proceso de requisitos iterativo en el cual se requiere una alta colaboración de los interesados y una planificación progresiva. Además, la documentación y aprobación de cambios será menos estricta.

Como se muestra en la Figura 2.2 la gestión de requisitos varía según el grado de predictibilidad o adaptabilidad del ciclo de vida del proyecto.

Figura 2.2. Gestión de requisitos según el ciclo de vida del proyecto



Nota: Adaptado de la guía *Requirements Management* (2016).

2.2.3 Gestión de requisitos y gestión de proyectos

Según PMI (2017), la gestión de requisitos mantiene una relación intrínseca con la gestión de proyectos. Además de estar fuertemente relacionada con la gestión del alcance del proyecto, la gestión de requisitos también lo está con la gestión de los interesados y las comunicaciones. Es por esto que debe existir una comunicación oportuna entre los encargados de la gestión de los requisitos y los encargados de la gestión del proyecto para garantizar una correcta integración de las actividades a realizar.

2.2.3.1 Gestión de requisitos y la gestión de los interesados

La gestión de los interesados es esencial para el éxito del proceso de gestión de requisitos, ya que mediante una proactiva y metodológica gestión de las necesidades y expectativas de los interesados se facilitará la obtención de los resultados esperados. La gestión de los interesados incluye las actividades para identificar a todas aquellas

personas y organizaciones impactadas por el proyecto, analizar sus características y desarrollar estrategias que fomenten su involucramiento.

2.2.3.2 Gestión de requisitos y la gestión de las comunicaciones

La gestión de las comunicaciones también es un factor clave para el éxito del proceso de gestión de requisitos, debido a que la información proveniente de los interesados debe ser correctamente comunicada al equipo del proyecto y viceversa. La gestión de las comunicaciones incluye las actividades requeridas para asegurar la apropiada planificación, recolección, creación, distribución, almacenamiento, gestión, control y monitoreo de la información del proyecto.

2.3 Especificaciones de requisitos de sistemas

Guide for Developing System Requirements Specifications es un documento del IEEE que brinda al lector una guía para identificar requisitos bien desarrollados y las formas de organizarlos. Esta guía establece que el desarrollo de especificaciones de requisitos de sistemas incluye la identificación, organización, presentación y modificación de los requisitos. A pesar de que esta guía no pretende ser un estándar para la industria, es mundialmente reconocida, por lo que fue utilizada como literatura base para desarrollar los conceptos de esta sección.

2.3.1 Generalidades

La colección de requisitos documentados y comunicados por parte del cliente hacia la comunidad técnica funge como medio de entendimiento, por lo que deben ser completamente entendibles por ambas partes. A esta colección se le conoce como especificación de requisitos.

2.3.1.1 Definición

La especificación de requisitos de sistemas representa el resultado de la definición de la necesidad, el concepto operacional y el análisis de las tareas a realizar. La representación de las especificaciones debe tener un formato adecuado para sus distintos interesados, por lo que suelen variar entre documentos escritos, modelos, prototipos o combinación de los anteriores.

2.3.1.2 Propiedades

La colección de requisitos debe contar con una serie de propiedades, las cuales se detallan a continuación:

- Conjunto único: cada requisito debe estar presente una sola vez.
- Normalizado: los requisitos no deben superponerse, haciendo referencia a algún otro.
- Conjunto enlazado: cada requisito debe estar ligado al resto mediante relaciones explícitas, que permitan comprender como en conjunto componen el sistema.
- Completo: la colección debe contener todos los requisitos expuestos por el cliente y aquellos necesarios para la definición del sistema.
- Consistente: la colección no debe presentar contradicciones en el nivel de detalle, el estilo de declaración de los requisitos ni en su presentación.
- Cerrado: deben identificarse los límites, el alcance y el contexto del conjunto de requisitos.
- Modificable: la colección debe ser modificable según sea necesario.
- Configurable: debe ser posible mantener versiones de la colección.
- Granular: la colección debe poder alcanzar el nivel de abstracción que permita definir el sistema.

2.3.1.3 Propósito

La especificación de requisitos de sistemas tiene como propósito describir las entradas y salidas del sistema, así como las interacciones entre ellas. Además, tiene como fin asegurar la correcta comunicación de estas entre el cliente y la comunidad técnica.

La especificación pretende ser comunicada de forma estructurada para asegurar que el cliente y la comunidad técnica sean capaces de desarrollar un entendimiento común, organizar los requisitos en niveles y verificar su completitud. Además, identificar los requisitos que se derivan de otros, las inconsistencias, las funcionalidades, condiciones y restricciones.

La especificación tiene el propósito de pactar acuerdos entre el cliente y la comunidad técnica. El cliente es un concepto colectivo que incluye al usuario a quién está destinado el funcionamiento del sistema, la organización solicitante, el ente que acepta el entregable y los gerentes que implementarán, operarán y mantendrán el sistema. La comunidad técnica incluye a los analistas, estimadores, diseñadores, aseguradores de la calidad, certificadores, desarrolladores, ingenieros, operadores y manufacturadores que participarán en el diseño, construcción y pruebas del sistema.

2.3.1.4 Beneficios

Si la especificación de requisitos está correctamente elaborada se obtendrán beneficios en las siguientes fases del ciclo de vida del proyecto. Por ejemplo:

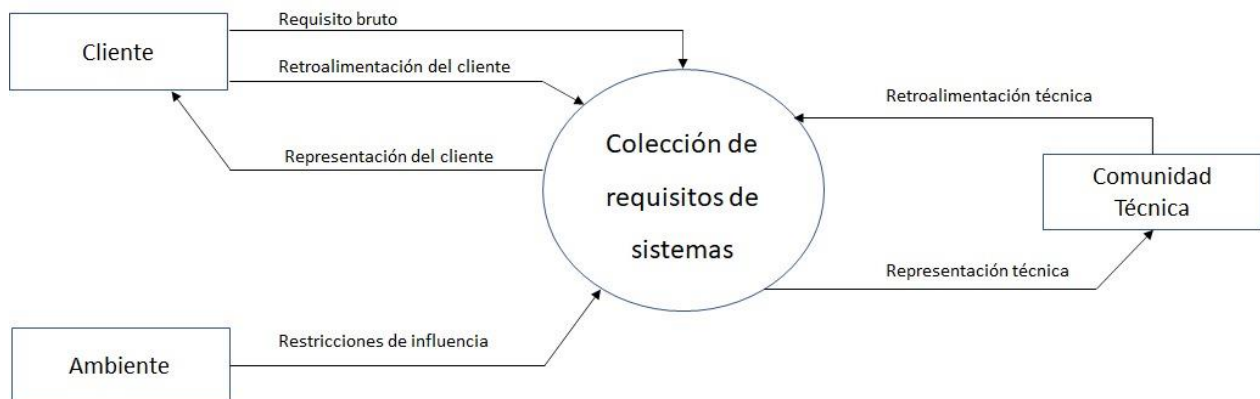
- Certeza para el cliente de que la comunidad técnica entiende la necesidad.
- Oportunidad temprana de retroalimentación bidireccional.
- Oportunidad de corregir problema y malentendidos a tiempo.
- Guía para la comunidad técnica para determinar cuando el sistema está completo.

- Soporte para los esfuerzos de planificación, diseño y desarrollo.
- Soporte en la gestión de cambios de requisitos.
- Protección contra malentendidos durante la ejecución.

2.3.2 Proceso

El contexto del desarrollo de requisitos de sistemas se compone de tres agentes externos y sus interacciones, como se muestra en la Figura 2.3 influyen el cliente, la comunidad técnica y el ambiente. En las siguientes secciones se detallarán estos agentes y sus interacciones.

Figura 2.3. Contexto de desarrollo de especificaciones de requisitos de sistemas



Nota: Adaptado de *Guide for Developing System Requirements Specifications* (1998).

2.3.2.1 Cliente

El cliente provee los objetivos, necesidades o problemas al proceso de desarrollo de requisitos de sistemas. Inicialmente, el cliente tiene un concepto impreciso que se conoce como requisito bruto, el cual puede ser documentado como:

- Concepto de operaciones: este documento se enfoca en los objetivos, metas y funciones generales deseados del sistema.

- Concepto de sistema: este documento incluye conceptos de información operacional, podría incluir diseños preliminares y requisitos explícitos.
- Especificación de mercadeo: este documento consiste en una lista de funcionalidades del sistema, define un tiempo de expectativa de puesta en producción e incluye un análisis de costo-beneficio.
- Solicitud de propuesta: este documento podría contener los documentos mencionados anteriormente con el propósito de solicitar asistencia para generar los documentos iniciales del sistema.
- Interfaces externas del sistema: este documento incluye las interfaces, límites y restricciones del universo externo al sistema. Un sistema puede tener distintas interfaces externas, como interfaces operacionales, eléctricas, enlaces de datos, protocolos y líneas de telecomunicación.

La representación al cliente incluye comunicación de la especificación de los requisitos del sistema para su clarificación o confirmación. La retroalimentación del cliente incluye la modificación de los objetivos, necesidad, requisitos actuales o la incorporación de nuevos requisitos.

2.3.2.2 Ambiente

El ambiente puede influir o generar restricciones en los requisitos del sistema, por lo que la identificación temprana y el monitoreo de los factores ambientales es clave para el éxito en el desarrollo de los requisitos del sistema. La influencia de estos factores puede provenir de aspectos políticos, de mercado, culturales y organizacionales, por lo que, tanto el cliente como la comunidad técnica deben estar

atentos para incorporar o adecuar los requisitos para satisfacer las nuevas condiciones forzadas por el ambiente.

2.3.2.3 Comunidad Técnica

La incorporación temprana de la comunidad técnica en el proceso de desarrollo de los requisitos facilita una detección temprana de nuevos requisitos y modificación de los requisitos existentes. La representación de la colección de requisitos por parte de la comunidad técnica pretende clarificar y confirmar los requisitos desde su criterio técnico experto. La retroalimentación técnica genera modificación, adición o eliminación de requisitos.

2.3.3 Requisitos bien formulados

Un requisito bien formulado es una declaración de una funcionalidad del sistema que puede ser validada, que resuelve un problema o alcanza un objetivo del cliente, que está calificada por condiciones mensurables y limitada por restricciones.

Para comprender si un requisito está correctamente formulado, es necesario diferenciar los conceptos de capacidad, condición y restricción. La capacidad es aquel requisito fundamental del sistema que representa una funcionalidad requerida por el cliente o necesaria para el sistema. La condición es un atributo de la capacidad, el cual es medible cuantitativa o cualitativamente. La restricción es un requisito impuesto a la solución por una circunstancia, fuerza o coacción, el cual impone límites que deben ser respetados. Estas restricciones pueden ser identificadas como requisitos independientes, como parte de una capacidad o enlazadas a múltiples requisitos.

2.3.3.1 Propiedades

Un requisito bien formulado cuenta con las siguientes propiedades:

- Abstracto: cada requisito debe ser implementado independientemente.
- Inequívoco: cada requisito debe ser interpretado de una única manera, sin dejar cabida a la ambigüedad.
- Rastreadable: para cada requisito debe ser posible determinar una relación con la necesidad establecida por el cliente.
- Validable: cada requisito debe disponer de los medios para demostrar que el sistema lo satisface.

2.3.3.2 Categorización

Al formular requisitos, se recomienda que sean categorizados por alguna de las siguientes maneras:

- Identificación: cada requisito debe ser identificado mediante un número o etiqueta único, de forma que sea posible relacionarlo con otros en caso de ser necesario.
- Prioridad: cada requisito debe contar con una prioridad definida por el cliente.
- Criticidad: cada requisito debe contar con un nivel de criticidad, ya que, un requisito podría tener una baja prioridad desde la perspectiva del cliente, pero sea indispensable para el funcionamiento del sistema.
- Factibilidad: cada requisito debe contar con un nivel de factibilidad, basado en el estado actual de la tecnología, ambiente del cliente y riesgo de elevación de costos.
- Riesgo: cada requisito debe contar con un grado de riesgo, relacionado con posibles pérdidas financieras, impacto ambiental, leyes o aspectos de salud y seguridad.
- Fuente: cada requisito debe establecer su originador, de forma que sea posible contactarlo para clarificar, modificar o eliminar dicho requisito.

- Tipo: cada requisito debe ser categorizado por uno o varios de los tipos:
 - Entrada
 - Salida
 - Fiabilidad
 - Disponibilidad
 - Mantenibilidad
 - Desempeño
 - Accesibilidad
 - Condiciones ambientales
 - Ergonomía
 - Seguridad
 - Requisito de instalación
 - Transportabilidad
 - Capacitación
 - Documentación
 - Interfaces externas
 - Pruebas

2.3.3.3 Errores en la formulación

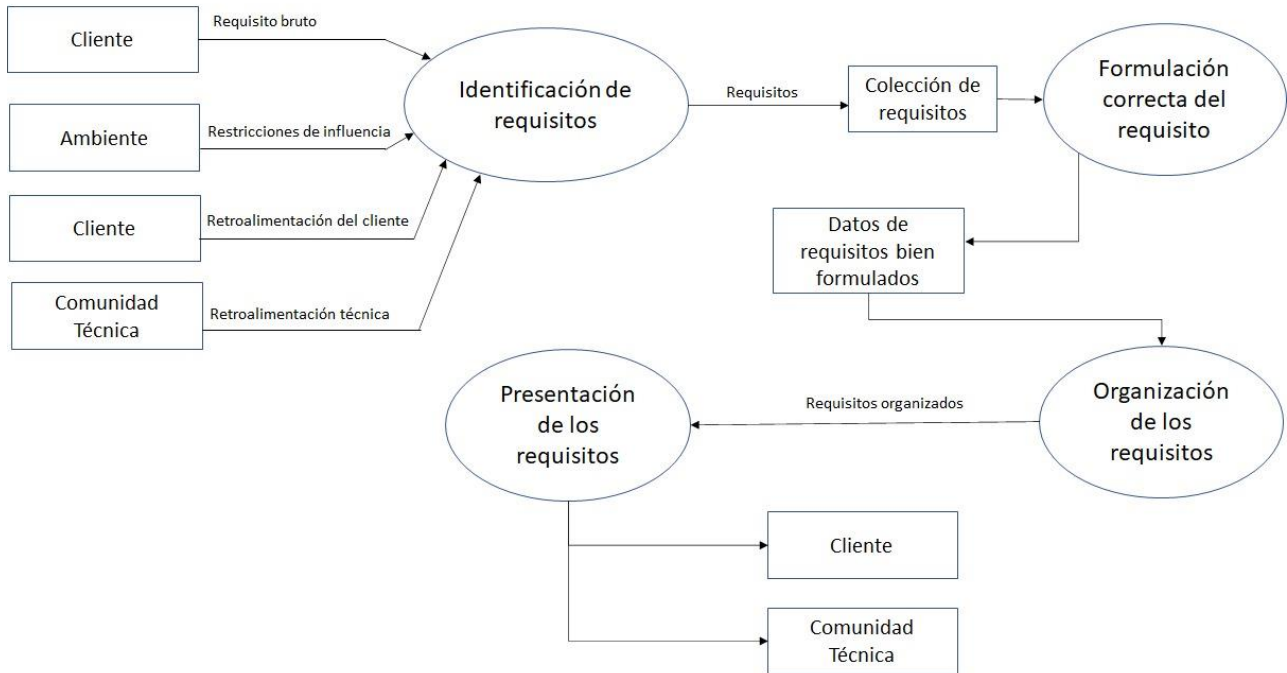
Algunos de los errores más comunes en la formulación de requisitos se presentan a continuación.

- Diseño e implementación: se suelen definir requisitos para incluir decisiones de diseño o implementación. Esta información debe ser documentada y comunicada de forma independiente.
- Sobre especificaciones:
 - Requisitos que expresan el funcionamiento exacto de un sistema comercial que puede ser comprado en lugar de creado.
 - Requisitos que establecen tolerancias para elementos profundos del sistema.
 - Requisitos que implementan soluciones.
- Restringidos: requisitos con restricciones innecesarias.
- Ilimitados: se suelen definir requisitos con declaraciones subjetivas, ambiguas o que no cuentan con límites definidos.
- Suposiciones: se definen requisitos basados en suposiciones no documentadas o relacionadas a situaciones futuras inciertas.

2.3.4 Desarrollo de requisitos

El proceso iterativo de desarrollo de especificaciones de requisitos de sistemas se compone de cuatro subprocesos, los cuales serán explicados en esta sección. Esta descomposición del proceso en subprocesos facilita el progreso completo y preciso de las especificaciones. En la Figura 2.4 se muestra el proceso de desarrollo de especificaciones de requisitos de sistemas, los agentes que interactúan en el proceso y sus interacciones.

Figura 2.4. Proceso de desarrollo de especificaciones de requisitos de sistemas



Nota: Adaptado de *Guide for Developing System Requirements Specifications* (1998).

2.3.4.1 Identificación de requisitos

La identificación de requisitos consiste en extraer de los clientes, la comunidad técnica y el ambiente un conjunto de requisitos. Este subproceso iterativo tiene como propósito solicitar todos los requisitos del sistema, asegurar que no falta ninguno y que cada uno es definido una única vez.

El plan para la identificación de requisitos debe ser comunicado al cliente y debe contener las técnicas y herramientas a utilizar. Algunas de las técnicas que pueden ser utilizadas para la identificación de los requisitos están los talleres de trabajo estructurados, la lluvia de ideas, las entrevistas, las encuestas, la observación, la revisión de documentación técnica, el análisis de mercado, la ingeniería inversa y las simulaciones.

2.3.4.2 Formulación correcta del requisito

La formulación correcta consiste en asegurar que cada requisito es necesario, corto, comprobable y que representa una necesidad. Además, pretende validar que se definan las condiciones adecuadas para cada requisito, evitando los errores comunes durante la formulación.

El subproceso debe asegurar también que los requisitos utilicen una redacción simple, con relaciones uniformes, definiciones únicas de palabras, símbolos y notaciones, así como el uso gramatical correcto del lenguaje y la simbología.

2.3.4.3 Organización de los requisitos

La organización de los requisitos consiste en crear una estructura para el conjunto de requisitos estableciendo las relaciones entre ellos. Durante este subproceso se buscan patrones entre los requisitos y se definen sus propiedades y atributos. La especificación de los requisitos de sistemas debe indicar la relación entre los requisitos, siendo la estructura jerárquica de capacidades el esquema más utilizado. En este esquema las capacidades más generales son descompuestas en requisitos de menor nivel.

2.3.4.4 Presentación de los requisitos

La presentación consiste en determinar las mejores maneras de comunicar los requisitos a cada interesado que requiera entender, revisar, aceptar o utilizar las especificaciones. Un solo método de representación no es siempre funcional, ya que los interesados usualmente tienen diferentes culturas y lenguajes o la representación de cierta información podría ser demasiado compleja en un método específico.

Algunos de los métodos que podrían ser utilizados para diferentes interesados son:

- Textual: papel o electrónico.
- Modelos: físicos, simbólicos, gráficos o prototipos.

La documentación de los requisitos aprobados usualmente requiere modificaciones, ya que puede contener errores de omisión, distorsiones o la generación de nuevos requisitos, por lo que controlar formalmente estos cambios es un factor crítico de la gestión de requisitos.

2.4 Especificaciones de requisitos de software

Recommended Practice for Software Requirements Specifications del IEEE (1998) describe las prácticas recomendadas para la creación de buenas especificaciones de requisitos de *software*. Estas prácticas tienen como objetivo facilitar al cliente describir con certeza los que desea obtener y al proveedor de *software* entender exactamente lo que el cliente desea. Este documento será utilizado como base teórica para exponer los elementos de esta sección.

2.4.1 Componentes de una especificación de requisitos de software

Una especificación de requisitos de *software* debe contener algunos componentes clave para considerarse adecuada. El nombre de estos componentes puede variar, pero deben ser incluidos para considerar que la especificación está completa. A continuación, se detalla cada componente.

2.4.1.1 Introducción

La introducción de la especificación de requisitos debe contar con una introducción que incluya:

- Propósito: describe el propósito de la especificación y su público objetivo.
- Alcance: identifica el nombre del producto de *software*, su funcionalidad, beneficios, objetivos y metas.
- Definiciones: define términos, acrónimos y abreviaturas para su correcta interpretación.
- Referencias: provee una lista completa de los documentos referenciados, incluyendo su título, fecha y fuente.
- Descripción: describe el contenido y organización del resto del documento.

2.4.1.2 Descripción general

Este componente describe los factores generales relacionados al producto de software y sus requisitos. Esta descripción se compone de seis elementos:

- Perspectiva del producto: describe el producto de *software* desde la perspectiva del sistema completo. La utilización de un diagrama de bloques puede apoyar a su comprensión. En este diagrama deben incluirse las interfaces de usuario, del sistema, de hardware, de software y de comunicación.
- Funciones del producto: provee un resumen de las funciones principales del software, usualmente tomadas de la sección de alto nivel de la especificación. Las funciones deben ser organizadas y representadas de forma textual o gráfica.
- Características del usuario: describe las características que debe poseer el usuario del software, incluyendo su nivel académico, experiencia y conocimiento técnico.
- Restricciones: describe las restricciones que limitan el accionar de los desarrolladores, incluyendo políticas regulatorias, limitaciones de hardware,

interfaces, operaciones paralelas, funciones de auditoría, funciones de control, requisitos de lenguaje, protocolos y consideraciones de seguridad.

- Supuestos: lista aquellos factores externos que se asume afectarán los requisitos estipulados.
- Prorratio de requisitos: identifica requisitos que deberán esperar hasta que estén disponible nuevas versiones del sistema.

2.4.1.3 Especificación del requisito

Este componente contiene todos los requisitos de software con el nivel de detalle suficiente para permitir un diseño que los satisfaga y unas pruebas que lo verifiquen. Estos requisitos deberán incluir al menos una descripción de cada entrada al sistema, cada salida desde el sistema y las funciones ejecutadas por el sistema. Los requisitos están compuestos por los siguientes elementos:

- Interfaces externas: descripción de todas las entradas y salidas del sistema de software. Debe incluir el nombre del ítem, la descripción de sus propósitos, las fuentes de entradas, los destinos de salida, las unidades de medición y los formatos de pantalla, ventana, datos y comandos.
- Funciones: descripción de las acciones que realizará el sistema mediante el procesamiento de entradas y la generación de salidas. Debe incluir validación de las entradas, secuencia de operaciones, respuestas a situaciones anormales, efecto de parámetros, relaciones entrada-salida y fórmulas de conversión entrada a salida.
- Requisitos de desempeño: especificación numérica de las capacidades del sistema.

- Requisitos de base de datos: especificación de tipos de información de la base de datos, frecuencia de uso, funciones de accesibilidad, entidades de datos, restricciones de identidad y retención de datos.
- Restricciones de diseño: define restricciones de diseño impuestas por estándares o limitaciones de otras índoles.
- Atributos del sistema de software: descripción de los atributos que permitan su verificación. Estos atributos pueden ser la fiabilidad, disponibilidad, seguridad, mantenibilidad y portabilidad.
- Organización de los requisitos: ordenamiento óptimo de los requisitos para su entendimiento. Existen múltiples formas de organizar los requisitos, entre ellas:
 - Modo de sistema: organización según el modo de operación del sistema. Por ejemplo, modo de entrenamiento, modo normal y modo de emergencia.
 - Clase de usuario: organización según las funcionalidades para cada usuario. Por ejemplo, pasajero y conductor.
 - Objetos: organización según el objeto al que está dirigido cada requisito. Por ejemplo, computador, escáner, impresora y caja registradora.
 - Funcionalidades: organización según funcionalidad. Por ejemplo, llamada de voz, video llamada y conferencia.
 - Estímulos: organización según estímulos que fungen como entradas del sistema. Por ejemplo, pérdida de poder y sobrecalentamiento.
 - Respuestas: organización según la respuesta generada por el sistema. Por ejemplo: generación de pagos, generación de reportes y notificación a usuarios.

- Jerarquía funcional: organización basada en una jerarquía de funcionalidades. Los requisitos son agrupados por entradas o salidas comunes.

2.4.1.4 Información complementaria

Se deben incluir tablas de contenidos, índices y apéndices que fortalezcan el formato de la especificación de requisitos de software. Los principales apéndices por incluir son ejemplos de formatos de entrada-salida, estudios de análisis de costos, resultados de encuestas a los usuarios, información complementaria que ayude al lector o una descripción de la problemática a resolver.

2.4.2 Consideraciones para formular especificaciones de requisitos

Una especificación de requisitos de *software* bien formulada debe considerar múltiples aspectos durante su redacción. Estos aspectos se exponen a continuación.

2.4.2.1 Naturaleza

La especificación de requisitos es elaborada para un producto de software particular que desarrolla ciertas funciones en el ambiente en que será implementado. Para formular una correcta especificación de requisitos, se debe documentarse la siguiente información:

- Funcionalidad: describe lo que se espera que el software realice.
- Interfaces externas: describe las interacciones que tendrá con las personas, *hardware* y otros *softwares*.
- Desempeño: describe condiciones como velocidad, disponibilidad, tiempo de respuesta y tiempo de recuperación.

- Atributos: describe cuales son las consideraciones de portabilidad, corrección y mantenibilidad.
- Restricciones de diseño: describe los estándares, políticas y limitaciones.

2.4.2.2 Ambiente

El software puede contener todas las funcionalidades de la solución, pero usualmente no es así, ya que forma parte de un sistema mayor. Por esto, deben considerarse los distintos componentes del sistema y sus interfaces, además de los requisitos originados por características del proyecto. Para formular una correcta especificación de requisitos, se debe tomar en cuenta que:

- Deben ser correctamente definidos todos los requisitos de software, ya sea originados por su naturaleza o debido a características del proyecto.
- No deben describirse detalles relacionados al diseño o implementación, ya que estos elementos forman parte de la planificación del proyecto y no del software.
- No deben imponerse restricciones adicionales al software.

2.4.2.3 Características

Una correcta especificación de requisitos de software debe contar con las siguientes características:

- Correcta: cada requisito establecido debe estar presente en el software, para garantizar que se satisface la necesidad.
- Inequívoca: los requisitos no presentan ambigüedad y son interpretados siempre de la misma manera.
- Completa: debe incluir todos los requisitos significativos relacionados a la funcionalidad, desempeño, restricciones de diseño, atributos e interfaces externas.

También debe incluir la definición de las respuestas del software a toda clase de entrada en cualquier situación.

- **Consistente:** los requisitos no presentan conflictos entre sí, están priorizados por importancia, necesidad o estabilidad.
- **Verificable:** debe ser posible verificar que el producto de software cumpla con lo establecido en el requisito, evitando términos subjetivos para describir dicha verificación.
- **Modificable:** cada requisito permite realizar cambios de forma sencilla, completa y consistente manteniendo su estructura y estilo.
- **Rastreado:** el origen de cada requisito es claro y facilita referenciarlo en desarrollos futuros o en la documentación.

2.4.2.4 Preparación conjunta

El proceso de desarrollo de software inicia con el acuerdo entre el cliente y el proveedor de lo que debe desarrollarse. Este acuerdo debe ser preparado en conjunto en la forma de una especificación de requisitos de software. Esto es de suma importancia, ya que, usualmente el cliente no comprende aspectos del diseño o desarrollo de software y el proveedor no comprende el problema y campo de acción del cliente.

2.4.2.5 Evolución

La especificación de requisitos de software debe evolucionar conforme avanza el proceso de desarrollo. Los requisitos deben ser especificados en su totalidad, aun sabiendo que cambiarán a futuro debido a su evolución y los cambios deben ser identificados, controlados y rastreados mediante un proceso formal de cambio.

2.4.2.6 Prototipos

La utilización de prototipos para el desarrollo de la especificación de requisitos de software es sumamente útil, ya que el cliente tiene la oportunidad de brindar su retroalimentación temprana gracias a su interacción con el prototipo. También, el prototipo permite observar aspectos no contemplados del comportamiento del software y se generan menos cambios en la especificación.

2.4.2.7 Diseño embebido

Al redactar una especificación de requisitos de software debe distinguirse con claridad la diferencia entre identificar restricciones de diseño y proyectar un diseño específico, ya que los requisitos generan limitaciones a las alternativas de diseño, pero no estipulan un diseño específico.

2.4.2.8 Requisitos del proyecto embebidos

Una especificación de requisitos de software debe incluir lo relacionado al producto de software y excluir lo relacionado al proceso de desarrollo. Los requisitos del proyecto representan el acuerdo entre el cliente y el proveedor acerca del proceso de desarrollo del producto, el cual incluye usualmente el costo, entregables, métodos de desarrollo de software, aseguramiento de la calidad, criterios de verificación y procedimientos de aceptación. Todos estos requisitos de proyecto deben ser documentados en otros documentos y no formar parte de la especificación.

2.5 Software de gestión *Jira*

Jira es un software propiedad de la compañía Atlassian, utilizado principalmente por equipos ágiles para la gestión del proceso de desarrollo de *software*, pero sus funcionalidades le permiten extender su alcance a diversas metodologías de gestión de proyectos. Inicialmente, Jira

fue ideado como un *software* de seguimiento de proyectos y tareas, pero ha evolucionado gracias a sus funciones que permiten planificar, dar seguimiento, controlar las entregas y generar reportes.

Atlassian, en su página web *Using Jira for Requirements Management* (2018), establece que la creación de flujos de trabajo personalizados, la integración con herramientas de terceros y la conexión con el mapa de ruta del producto, facilitan la gestión de requisitos con Jira.

2.5.1 Scrum con Jira

La gestión de los proyectos bajo el marco de trabajo *Scrum* puede ser llevada a cabo mediante las herramientas que provee Jira. Según Atlassian y su página web *How to do Scrum with Jira software* (2018), se definen los siguientes pasos a seguir para lograrlo:

1. Crear un proyecto *Scrum*: después de iniciar sesión, se debe crear un nuevo proyecto de tipo *Scrum*, lo que generará un *product backlog* vacío.
2. Crear historias de usuario o tareas: se deben crear y priorizar las historias de usuario y tareas en el *product backlog*.
3. Crear un *sprint*: se debe crear el primer *sprint*.
4. Realizar el *sprint planning*: se debe realizar la sesión de *sprint planning* con el equipo para seleccionar y descomponer las historias de usuario. Además, de estimar el esfuerzo para ejecutar cada historia de usuario y tarea.
5. Iniciar el *sprint* en Jira: se debe nombrar el *sprint* y definir sus fechas de inicio y fin. Una vez iniciado el *sprint*, se habilitará la sección *active sprint* y el tablero *Scrum* correspondiente, en la cual el equipo tiene la capacidad de mover las tareas de la columna por-hacer a la columna en-progreso y eventualmente a la columna hecho.
6. Realizar el *daily Scrum*: se debe realizar el *daily Scrum* visualizando el tablero *Scrum* para comunicar el progreso del trabajo durante el *sprint*.

7. Ver el reporte del *sprint*: durante o al finalizar el *sprint*, es posible observar el reporte del *sprint*.
8. Realizar la sesión de *sprint review*: se debe realizar la sesión de *sprint review* para mostrar a los interesados la entrega realizada durante el *sprint*.
9. Realizar la sesión de *sprint retrospective*: se debe realizar la sesión de *sprint retrospective*. La documentación de esta sesión puede ser realizada en Confluence, una herramienta de *software* complementaria de Jira.
10. Completar el *sprint*: Al finalizar el *sprint*, se debe completar el *sprint* en Jira. Al ejecutar esta acción las tareas que no lograron ser completadas pueden regresar al *backlog* o ser asignadas al siguiente *sprint*.
11. Repertir desde el paso 2.

2.5.2 Hojas de ruta

Según Atlassian y su página web Hojas de ruta de productos (2018), Jira permite la creación de hojas de ruta del producto, las cuales consiste en un plan de acción de cómo un producto evoluciona a lo largo del tiempo. Los *Product Owners* utilizan las hojas de ruta para describir las futuras funcionalidades del producto y sus entregas. La hoja de ruta es un insumo valioso para el equipo *Scrum* durante sus sesiones de planificación y adaptación, y que le permite tener una visión general del producto y facilitar la toma de decisiones.

La utilización de hojas de ruta del producto genera beneficios para la gestión de proyectos y la gestión de requisitos:

- Los administradores de proyectos obtienen actualizaciones del estado del trabajo y e interpretación del trabajo en Jira desde una perspectiva no técnica.

- El *Product Owner* obtiene una visión general del trabajo realizado en un mismo producto por distintos equipos y facilita la comunicación de las prioridades.
- Los desarrolladores obtienen una mejor idea del producto esperado, pueden centrarse en las tareas más importantes, evitan la corrupción del alcance y toman mejores decisiones.

2.5.3 *Confluence*

El software Confluence, propiedad de Atlassian, se integra sin problemas con Jira, permitiendo dar seguimiento de los requisitos del proyecto en Jira desde la documentación en Confluence.

Según Atlassian y su página web *Plantilla para los requisitos de productos* (2018), Confluence cuenta con una plantilla destinada a la gestión de los requisitos, la cual pretende concretar los requisitos del producto con el equipo de desarrollo, en busca de asegurar de que los requisitos hayan sido bien desarrollados y probados. Se establecen algunos pasos a seguir para utilizar esta plantilla:

- *Abarcar lo básico*: establecer los detalles del producto, su fecha de entrega prevista, su estado actual y los responsables del equipo.
- *Trazar el objetivo y las métricas*: exponer los objetivos del producto y las métricas que se utilizarán para validar el éxito del producto.
- *Enumerar las suposiciones*: escribir cualquier suposición sobre los usuarios, limitaciones técnicas y los objetivos comerciales.
- *Añadir documentación de soporte*: incluir toda la documentación de soporte relacionada a los requisitos del producto, entre las cuales destacan las maquetas, diagramas o diseños visuales.

Capítulo 3. Marco metodológico

En este capítulo se expone la metodología utilizada para desarrollar la investigación, lograr los entregables de los objetivos específicos y la solución a la problemática planteada. Además, se definen las categorías de estudio, las técnicas e instrumentos para la recolección de datos, se identifican los sujetos y fuentes de información, se describen los productos a obtener de cada objetivo específico y se definen los métodos y herramientas a utilizar en la etapa de análisis y procesamiento.

Esta investigación tiene un enfoque cualitativo. Según Niño, V. (2011) la investigación cualitativa consiste en recolectar y analizar la información de diversas formas, sin traducirse en términos matemáticos, tiende a centrarse en casos que ejemplifican la problemática y su meta es lograr profundidad.

3.1 Categorías de la investigación

Para poder comprender los factores que rodean la problemática expuesta y explorar sus posibles soluciones, se define una categoría de investigación compuesta de subcategorías. Estas son características sujetas a la observación y análisis, por medio de las cuales es posible obtener la información deseada.

En el Cuadro 3.1, es posible observar la estructura de categoría y subcategorías utilizada para esta investigación. La investigación se centra en una única categoría que engloba los elementos de la gestión de requisitos de proyectos, los cuales consisten en las actividades para identificar los requisitos de un proyecto, establecer su línea base, trazabilidad y control de cambios. Además, se definen dos subcategorías asociadas a los dos primeros objetivos específicos planteados.

La primera subcategoría consiste en la investigación de las prácticas ejecutadas actualmente por los equipos *Scrum* de la Tribu Navi para la gestión de requisitos de proyectos. Esta subcategoría pretende determinar la forma en que se realiza actualmente la gestión de requisitos de proyectos dentro del marco de trabajo *Scrum*, identificar aquellos elementos que tienen relación con la problemática identificada en la Tribu Navi y validar si el personal cuenta con las competencias necesarias para gestionar los requisitos. Para lograrlo, se utiliza la técnica de revisión documental para la extracción de información de los documentos internos de la Tribu Navi, la técnica de grupo focal para la obtención de información por parte de los miembros de cada equipo *Scrum* y la técnica de entrevista para profundizar en la temática con los *Scrum Master*.

La segunda subcategoría consiste en la investigación de las prácticas consideradas adecuadas globalmente para la gestión de requisitos de proyectos. Esta subcategoría pretende generar un compendio de buenas prácticas asociadas a la gestión de requisitos y determinar cómo estas prácticas solventan los elementos relacionados a la problemática. Para esto, se utilizará la técnica de revisión documental en busca de hallar las buenas prácticas en la literatura y la técnica de análisis de casos de referencia para considerar las prácticas adoptadas por distintas organizaciones reconocidas mundialmente.

Cuadro 3.1. Categorías de la investigación

Categoría:	Gestión de requisitos de proyectos			
Definición conceptual de la categoría:	Actividades para desarrollar y gestionar los requisitos de un proyecto, que permiten identificar los requisitos, establecer su línea base, trazabilidad y control de cambios.			
Sub-categoría	Definición Conceptual	Pregunta Generadora	Técnicas	Instrumentos
1. Prácticas actuales de gestión de requisitos de proyectos utilizadas por los equipos <i>Scrum</i> de la Tribu Navi.	Procesos, técnicas y herramientas utilizadas actualmente por los equipos <i>Scrum</i> para la gestión de requisitos de proyectos.	<p>¿Cómo se realiza actualmente la gestión de requisitos de proyectos con <i>Scrum</i>?</p> <p>¿Cuáles elementos de las prácticas actuales influyen en la problemática?</p> <p>¿Cuáles competencias para la gestión de requisitos están presentes en los equipos <i>Scrum</i>?</p>	Revisión documental	Ficha de síntesis
				Inventario
				Lista de verificación
			Grupo Focal	Cuestionario
			Entrevista	Guía de Entrevista
2. Buenas prácticas de gestión de requisitos de proyectos	Procesos, técnicas y herramientas consideradas adecuadas globalmente para la gestión de requisitos de proyectos.	<p>¿Cuáles son las buenas prácticas de gestión de requisitos de proyectos?</p> <p>¿Cómo se relacionan estas prácticas con los elementos de la problemática?</p>	Revisión documental	Ficha de síntesis
				Inventario
			Análisis de casos de referencia	Ficha de síntesis
				Inventario

3.2 Etapa de recolección de información

En esta sección se detallan los sujetos y fuentes de los que se obtienen los datos e información, así como las técnicas e instrumentos que se aplican a dichos sujetos y fuentes.

3.2.1 Sujetos de información

Según Niño V. (2011) los sujetos de información son aquellos de los cuales la persona investigadora extrae información de valor, gracias a su contacto directo con la realidad de la problemática. Hernández, R. (2014) establece que, en una investigación cualitativa, los datos que interesan son conceptos, percepciones y pensamientos manifestados por los participantes, ya sea de manera individual, grupal o colectiva.

Por lo que, debido a la naturaleza colectiva de la problemática, son definidos como sujetos de información cada uno de los equipos *Scrum* de la Tribu Navi y no sus individuos. Como se muestra en el Cuadro 3.2, los equipos *Scrum* OMA, TAW, KEK e IKN serán sujetos de información. La investigación abarca toda la población, ya que incluye a los cuatro equipos *Scrum* de la Tribu Navi. Además, el *Scrum Master* es un sujeto de información independiente, debido a su amplio conocimiento de la metodología de gestión de proyectos actual. La investigación abarca la población de dos *Scrum Master* de la Tribu Navi.

Cuadro 3.2. Sujetos de información

Sujeto	Rol del sujeto	Información por obtener
Equipo OMA	Equipo de la Tribu Navi con más de cuatro años de experiencia en el marco de trabajo <i>Scrum</i> , conformado por un <i>Product Owner</i> , un <i>Scrum Master</i> y los <i>Developers</i> .	- Prácticas ejecutadas por los equipos <i>Scrum</i> para la gestión de requisitos de proyectos. - Competencias para la gestión de requisitos de proyectos - Elementos relacionados con la problemática.
Equipo TAW		
Equipo KEK		
Equipo IKN		
<i>Scrum Master</i>	Profesional en tecnologías de información con capacitación formal en el marco de trabajo <i>Scrum</i> y experiencia de al menos dos años en el puesto.	

3.2.2 Fuentes de información

Según Niño V. (2011), las fuentes de información son documentos escritos o plasmados por otro medio que proporcionan información de valor para la investigación. En el Cuadro 3.3, se detallan las fuentes utilizadas para la recolección de información de esta investigación, las cuales están clasificadas por temática y tipo de fuente. El tipo de fuente primaria es aquella que está directamente relacionada con los tópicos de la investigación y la secundaria brinda información complementaria que sustenta la fuente primaria.

Las fuentes con temática relacionada a la metodología de proyectos permitieron comprender los principios de gestión ágil de proyectos y conceptos relacionados al marco de trabajo *Scrum*. Las fuentes primarias de esta temática son la Guía de *Scrum*, la *Agile Practice Guide* del PMI y la Guía PMBok® del PMI.

Las fuentes relacionadas a la gestión de requisitos permitieron identificar las prácticas actuales y las buenas prácticas acordes al marco de trabajo *Scrum*. Entre ellas se encuentran el Manual Operativo *Scrum* de la Tribu Navi y la documentación en Jira, las cuales tienen como objetivo brindar una perspectiva de las prácticas actuales de gestión de requisitos de proyectos utilizadas por los equipos *Scrum*. También, se incluyeron fuentes bibliográficas primarias como la guía *Requirement Management* del PMI y guías de especificación de requisitos del IEEE para la identificación de las buenas prácticas. Además, las fuentes secundarias estuvieron destinadas a brindar información complementaria acerca de las buenas prácticas.

Cuadro 3.3. Fuentes de información

Temática	Fuentes	Tipo de Fuente	Información por obtener
Metodología de Proyectos	Schwaber, K. y Sutherland, J. (2020). <i>La Guía de Scrum</i> .	Primaria	Principios de gestión ágil de proyectos.
	Project Management Institute (2017). <i>Agile Practice Guide</i> .	Primaria	
	Project Management Institute (2017). <i>A guide to the project management body of knowledge PMBOK® Sixth edition</i> .	Primaria	
Gestión de Requisitos	Manual Operativo <i>Scrum</i> de la Tribu Navi	Primaria	Prácticas de gestión de requisitos utilizadas por los equipos <i>Scrum</i> .
	Documentación en Jira de la Tribu Navi.	Primaria	
	Project Management Institute (2016). <i>Requirement Management: A Practice Guide</i> .	Primaria	Buenas prácticas de gestión de requisitos de proyectos acordes al marco de trabajo <i>Scrum</i> .
	Cohn, M. (2004). <i>User Stories Applied for Agile Software Development</i> . Pearson Education. Boston, Massachusetts.	Primaria	
	Atlassian (2021), <i>Agile Coach</i> .	Primaria	
	NASA (2007), <i>Systems Engineering Handbook</i> . NASA Headquarters, Room 6D37. Washington, DC.	Secundaria	
	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i> . Rockwell Collins, Inc. Washington, DC.	Secundaria	
Infosys (2019), <i>Model-Based Systems Engineering: Digitized Requirements Management for Aerospace and Defense Products</i> . Bengaluru, India.	Secundaria		

3.2.3 Técnicas de recopilación de datos

Según Niño V. (2011) las técnicas de recopilación de datos son procedimientos que se aplican en la investigación para recoger la información o los datos requeridos. Las técnicas de recopilación tienen como objetivo recolectar, conservar, analizar y transmitir los datos sobre los cuales se investiga. En esta sección se exponen las técnicas utilizadas para esta investigación.

3.2.3.1 Grupo focal

Se utilizó la técnica de grupo focal para identificar las prácticas de gestión de requisitos utilizadas por los equipos *Scrum*. Se realizó un grupo focal vía conferencia virtual con cada equipo *Scrum*, con el objetivo de que los miembros intercambiaran sus opiniones y llegaran a un consenso para brindar información clara y consistente como equipo. Se optó por esta técnica debido a la compatibilidad que presenta con respecto a las sesiones realizadas a diario por los equipos *Scrum*.

Cada grupo focal tuvo una duración máxima de 90 minutos, fue dirigido por el *Scrum Master* correspondiente y estuvo conformado por todos los miembros del equipo *Scrum*, incluyendo al *Scrum Master*, *Product Owner* y *Developers*.

Durante el grupo focal, se aplicó el instrumento de cuestionario de diagnóstico que se muestra en el apéndice A para identificar la frecuencia con la que se ejecutan las prácticas específicas de gestión de requisitos, así como las competencias del personal para ejecutarlas. Las respuestas obtenidas fueron ingresadas al cuestionario por el *Scrum Master*.

3.2.3.2 Entrevista

Se utilizó la técnica de entrevista para obtener un panorama general de los procesos, técnicas y herramientas utilizadas por los equipos *Scrum*, incluyendo el flujograma actual de procesos para la gestión de requisitos.

La entrevista fue aplicada posteriormente a la obtención de los resultados del cuestionario de diagnóstico y consistió en una sesión virtual con duración máxima de 60 minutos, entre el entrevistador y los *Scrum Master* de la Tribu Navi. La entrevista utilizó el instrumento de guía de entrevista del apéndice B como insumo para el entrevistador.

3.2.3.3 Revisión documental

Para la primera subcategoría de la investigación, se utilizó la técnica de revisión documental con el fin de identificar las prácticas de gestión de requisitos de proyectos utilizadas actualmente por los equipos *Scrum*. Esta técnica fue aplicada para la revisión a profundidad del Manual Operativo *Scrum* de la Tribu Navi y de la documentación en Jira.

Se utilizó el instrumento de ficha de síntesis como se muestra en el apéndice C para extraer todas aquellas prácticas de gestión de requisitos identificadas y los elementos relacionados a la problemática. Se utilizó un instrumento de inventario como se muestra en el apéndice D para organizarlas y facilitar su comprensión. Además, se utilizó un instrumento de lista de verificación como se muestra en el apéndice E para identificar las prácticas de documentación de requisitos utilizadas por los equipos *Scrum* en Jira.

Para la segunda subcategoría de la investigación, se realizó una revisión documental para identificar las buenas prácticas de gestión de requisitos de proyectos

en la literatura. Para esto, se estudiaron a profundidad las fuentes primarias y secundarias del Cuadro 3.3 relacionadas a la gestión de requisitos.

Mediante fichas de síntesis, se extrajeron todas las prácticas de gestión de requisitos consideradas adecuadas según el marco de trabajo *Scrum*, con la finalidad de tener un concepto claro y resumido de cada práctica. Se incluyó dentro de cada ficha una sección que explica brevemente cómo la práctica solventa los elementos relacionados a la problemática. Además, se utilizó un inventario para organizar las prácticas identificadas y así poder ser fácilmente revisadas durante la formulación de la propuesta a la problemática.

3.2.3.4 Análisis de casos de referencia

Se optó por la técnica de análisis de casos de referencia para identificar buenas prácticas de gestión de requisitos en organizaciones reconocidas mundialmente y así poder considerarlas como parte de la propuesta de solución a la problemática.

Se analizaron tres casos de organizaciones que cuentan con prácticas documentadas y estandarizadas para la gestión de requisitos. Se utilizaron las fichas de síntesis para extraer concretamente los conceptos de las prácticas y su aporte a la resolución de la problemática y se creó un inventario para organizar la información.

3.2.4 Instrumentos de recopilación de datos

La investigación contempló la utilización de distintos instrumentos de recopilación de datos, A continuación, se detalla cada uno de ellos.

3.2.4.1 Cuestionario de diagnóstico

El cuestionario de diagnóstico que se muestra en el apéndice A tiene un formato digital y requiere de un computador con acceso a Internet para ser contestado, por lo que, durante el grupo focal, el *Scrum Master* es responsable de realizar y responder a las preguntas en consenso con el resto de su equipo.

Está compuesto por 36 preguntas, divididas en tres secciones. La primera sección, contiene 18 preguntas que pretenden identificar la frecuencia con la que se ejecutan las prácticas específicas de gestión de requisitos. La segunda sección, cuenta con ocho preguntas relacionadas a las prácticas de formulación de requisitos. Por último, la tercera sección cuenta con 10 preguntas que pretenden identificar si los equipos *Scrum* cuentan con conocimiento y experiencia en la gestión de requisitos.

Las preguntas utilizan una escala del uno al cinco, la cual debe ser interpretada de la siguiente manera:

1. Nunca
2. Casi nunca
3. Ocasionalmente
4. Casi siempre
5. Siempre

Para facilitar su aplicación, el cuestionario incluye una sección explicativa inicial y un glosario de términos.

3.2.4.2 Guía de entrevista

El instrumento de guía de entrevista que se muestra en el apéndice B tiene como objetivo servir como insumo al entrevistador durante la aplicación de la técnica de entrevista a los *Scrum Master* y cuenta con una sección explicativa inicial.

La guía de entrevista cuenta con cinco preguntas abiertas relacionadas a la gestión de requisitos de proyectos. Cada pregunta debe ser contestada ampliamente, haciendo énfasis en los detalles relacionados a los procesos, técnicas y herramientas utilizadas.

3.2.4.3 Ficha de síntesis

La ficha de síntesis consiste en un cuadro resumen de los datos más relevantes con respecto a la práctica de gestión de requisitos identificada. Su objetivo es brindar un concepto general de la práctica y su relación con la problemática.

Debe incluir un nombre e identificador único, una descripción general de la práctica identificada, un listado de los elementos de la problemática a los que se relaciona o solventa y los datos relacionada a la fuente de información de la cual fue extraída, como se muestra en el apéndice C.

La ficha podrá incluir material gráfico que permita una mejor comprensión de la práctica identificada.

3.2.4.4 Inventario

El inventario consiste en un cuadro resumen que pretende organizar y categorizar las prácticas identificadas y documentadas en las fichas de síntesis. Este instrumento brinda un panorama global de todas las prácticas identificadas y su relación con la problemática. Mediante el inventario, es posible observar de manera

unificada si existen varias prácticas relacionadas a un mismo elemento de la problemática o si existen prácticas alternativas para la propuesta de solución.

El inventario debe contar con el formato que se muestran en el apéndice D.

3.2.4.5 Lista de verificación

La lista de verificación es un instrumento utilizado durante la revisión documental de Jira que pretende verificar si la documentación actual de los requisitos incluye los componentes recomendados y si están bien formulados.

Primero, la lista permite verificar si las historias de usuario cumplen con los elementos de rol, expectativa y razón que la conforman. Segundo, permite validar si se cumple con el criterio INVEST, en el cual los requisitos deben ser independientes, negociables, valorables, estimables, pequeños y comprobables. Finalmente, permite validar si la documentación incluye información complementaria como representaciones gráficas, material visual de apoyo o material documental.

Además, la lista incluye una sección de comentarios dedicada para incluir cualquier elemento relacionado con la problemática y cuenta con el formato que se muestran en el apéndice E.

3.3 Etapa de procesamiento y análisis de datos

En esta sección se detallan los métodos que se utilizaron para el análisis y procesamiento de datos, así como, los productos obtenidos de la investigación y cómo satisfacen los objetivos específicos planteados.

3.3.1 Métodos para el procesamiento y análisis de datos

Para satisfacer el primer objetivo específico se planteó la obtención de dos productos de la investigación. El primero, un flujograma de procesos actuales de gestión de requisitos y el segundo, un inventario de prácticas actuales de gestión de requisitos y elementos que influyen en la problemática.

La elaboración del flujograma utilizó los siguientes métodos de procesamiento y análisis de datos:

- Síntesis de datos para reunir y organizar los procesos de gestión de requisitos identificados en la entrevista y en el Manual Operativo *Scrum*.
- Diagramación para representar en un flujograma dichos procesos.

El inventario de prácticas actuales utilizó los siguientes métodos de procesamiento y análisis de datos:

- Síntesis de datos para reunir y organizar las prácticas de gestión de requisitos identificadas en la entrevista y en el Manual Operativo *Scrum*.
- Codificación y tabulación de los datos de la lista de verificación.
 - Codificación: se asignaron valores de 1 a los elementos presentes en la lista de verificación y de 0 a los que no lo están.
 - Tabulación: se creó una tabla de sumatoria, en la cual cada fila corresponde a un requisito y cada columna a un elemento de la documentación. Se completó la tabla con los valores de 0 o 1 según la lista de verificación y se sumó cada columna para obtener su total.
- Organización y tabulación de los datos del cuestionario de diagnóstico.

- Organización: se establecieron tres secciones para la organización de los datos: gestión de requisitos, formulación de requisitos y competencias.
 - Tabulación: se creó una tabla para cada sección, en donde cada fila corresponde a una pregunta del cuestionario acompañada del resultado obtenido en la escala del 1 a 5 para cada equipo *Scrum*. Se sumaron los resultados obtenidos de cada equipo para obtener un estado general de la frecuencia de uso de cada práctica en la Tribu Navi. Se ordenaron las tablas de forma que las prácticas más utilizadas estuviesen en la parte superior y las menos utilizadas en la parte inferior.
- Triangulación y tabulación de datos obtenidos de las fichas de síntesis, entrevista, cuestionario y lista de verificación. Mediante la triangulación y tabulación de datos fue posible realizar el análisis de resultados y la creación del inventario.

Para satisfacer el segundo objetivo específico se plantearon dos productos. El primero, un inventario de buenas prácticas de gestión de requisitos y el segundo, un cuadro de análisis.

El inventario de buenas prácticas utilizó los siguientes métodos de procesamiento y análisis de datos:

- Organización de datos de las fichas de síntesis extraídas de las fuentes de información relacionadas a la gestión de requisitos según el elemento de la problemática que solventan.
- Tabulación: se tabularon las prácticas identificadas a partir de las fichas de síntesis para la creación del inventario. El inventario fue organizado según el elemento de la problemática que solventa cada práctica.

El cuadro de análisis utilizó los siguientes métodos de procesamiento y análisis de datos:

- Comparativa de los datos del inventario de prácticas actuales y el inventario de buenas prácticas para analizar los elementos de la problemática presentes y sus posibles soluciones.

El tercer objetivo específico fue satisfecho mediante el diseño y redacción de una guía de procesos, técnicas y herramientas. Además, de la diagramación de un flujograma de procesos actualizado con la incorporación de las buenas prácticas seleccionadas y la recolección y organización de términos para la producción de un diccionario de lenguaje común.

Para el cuarto objetivo específico, se realizó la formulación del cronograma, presupuesto y matriz de responsabilidades de la implementación mediante los métodos de diseño y redacción.

En el Cuadro 3.4 se muestra un resumen de los métodos de análisis y procesamiento de datos utilizados para obtener los productos esperados y satisfacer cada objetivo específico.

Cuadro 3.4. Métodos para el procesamiento y análisis de los datos

Objetivo Específico	Producto	Métodos
1. Identificar las prácticas de gestión de requisitos de proyectos utilizadas actualmente por los equipos <i>Scrum</i> mediante el análisis de los 12 <i>sprints</i> del estudio para la descripción de sus procesos, técnicas, herramientas y elementos que influyen en la problemática.	Flujograma de procesos actuales de gestión de requisitos.	<ul style="list-style-type: none"> - Síntesis de datos. - Diagramación de información.
	Inventario de prácticas actuales de gestión de requisitos y elementos que influyen en la problemática.	<ul style="list-style-type: none"> - Síntesis de datos - Organización de datos - Codificación de datos - Tabulación de datos - Triangulación de datos.
2. Exponer las buenas prácticas de gestión de requisitos de proyectos acordes al marco de trabajo <i>Scrum</i> mediante la revisión bibliográfica para ser incorporadas en los procesos, técnicas y herramientas actuales.	Inventario de buenas prácticas de gestión de requisitos.	<ul style="list-style-type: none"> - Organización de datos - Tabulación de datos
	Cuadro de análisis	<ul style="list-style-type: none"> - Comparativa de datos. - Selección de datos.
3. Rediseñar los procesos, técnicas y herramientas de gestión de requisitos de proyectos por medio de la incorporación de las buenas prácticas expuestas para la generación de un método estándar.	Guía de procesos, técnicas y herramientas	<ul style="list-style-type: none"> - Diseño y redacción.
	Flujograma de procesos actualizado con la incorporación de las buenas prácticas.	<ul style="list-style-type: none"> - Diagramación de información.
	Diccionario de lenguaje común	<ul style="list-style-type: none"> - Recolección y organización de datos.
4. Desarrollar una estrategia de implementación del método de gestión de requisitos de proyectos mediante la descripción de las actividades, presupuesto y responsables para la ejecución de la propuesta en la Tribu Navi.	Cronograma de la implementación	<ul style="list-style-type: none"> - Diseño y redacción.
	Presupuesto de la implementación	<ul style="list-style-type: none"> - Diseño y redacción.
	Matriz de responsabilidades de la implementación	<ul style="list-style-type: none"> - Diseño y redacción.

Capítulo 4. Análisis de Resultados

En este capítulo se presentan los resultados obtenidos de la aplicación de las técnicas e instrumentos de recolección y procesamiento de datos definidos en el marco metodológico para cada subcategoría de investigación.

4.1 Prácticas actuales de gestión de requisitos

En esta sección se exponen los resultados obtenidos de la revisión documental, grupo focal y entrevista, los cuales se muestran en las secciones 4.1.1, 4.1.2 y 4.1.3, respectivamente. Por su parte, el análisis de resultados conjunto es expuesto en la sección 4.1.4, el cual consiste en la triangulación de datos resultantes para exponer los hallazgos que fundamenten la propuesta de solución. Finalmente, se exponen los productos resultantes de la investigación, el flujograma de procesos y el inventario de prácticas actuales en las secciones 4.1.5 y 0, respectivamente.

4.1.1 Resultados de la revisión documental

Los resultados obtenidos de la revisión del Manual Operativo *Scrum* se presentan en formato de ficha de síntesis en el apéndice F y los extraídos de la revisión documental de Jira se muestran en el apéndice G.

En la Tabla 4.1, se muestra un resumen de los resultados obtenidos de la revisión documental de Jira. Esta revisión consiste en el estudio de 540 requisitos documentados en el software Jira, con el fin de determinar el porcentaje de aparición de cada uno de los elementos de documentación que son incluidos. El análisis de resultados y triangulación de datos correspondientes se muestran en la sección 4.1.4.

Tabla 4.1. Prácticas actuales de documentación de requisitos

Elemento verificado	Porcentaje de aparición
Rol	44.26%
Expectativa	48.89%
Razón	22.41%
Independiente	47.59%
Negociable	47.59%
Valorable	47.59%
Estimable	18.89%
Pequeño	21.48%
Testeable	19.07%
Representación gráfica	0.93%
Material visual de apoyo	12.41%
Material documental	6.11%

En la Tabla 4.1, se observa que los requisitos documentados incluyen el elemento de rol en un 44.26%, de expectativa en un 48.89% y de razón en un 22.41% de las ocasiones.

Los resultados obtenidos muestran que el concepto del rol en las historias de usuario no es comprendido correctamente por los equipos *Scrum*, ya que en el 20,74% de las ocasiones, no fueron capaces de formular los requisitos desde el punto de vista de un usuario real del producto, como se muestra en la sección de comentarios del apéndice G.

Un requisito bien formulado debe incluir un rol de usuario adecuado a la funcionalidad por desarrollar. Por ejemplo, si el requisito consiste en la creación de una funcionalidad de generación de reportes transaccionales desde la aplicación de punto de venta, algunos roles de usuario adecuados podrían ser el tarjetahabiente o el cajero del comercio afiliado.

Algunos ejemplos de los roles formulados inadecuadamente se muestran a continuación:

- Usuario de negocio
- Usuario de desarrollo
- *Product Owner*
- Desarrollador
- Equipo de desarrollo

Además, al realizar la verificación del método INVEST para la redacción de las historias de usuario se obtiene como resultado que los requisitos documentados son independientes, negociables y valorables en un 47.59%, son estimables en un 18.89%, pequeños en un 21.48% y testeables en un 19,07% de las ocasiones.

Por otra parte, se identifica que la redacción de la expectativa en las historias de usuario fomenta la ambigüedad y dificulta establecer un estimado correcto del esfuerzo, ya que se utilizan términos genéricos como prueba, homologación o propuesta sin especificar con claridad lo que se desea obtener con dicha historia de usuario. Algunas de las expectativas mal redactadas se muestran a continuación:

- "Realizar una prueba integral"
- "Avanzar los riesgos tecnológicos identificados"
- "Realizar pruebas controladas"
- "Completar la documentación"
- "Trabajar en la propuesta"
- "Realizar la homologación"
- "Validar el diseño y funcionalidad del módulo"

Finalmente, los requisitos documentados incluyeron representaciones gráficas como diagramas o esquemas en un 0,93%, material visual de apoyo como imágenes o videos en un 12.41% y material documental como guías o manuales en un 6.11% de las ocasiones. Esto evidencia que los requisitos documentados en Jira no se acompañan de material de apoyo que los sustenten, valiéndose principalmente de la redacción del enunciado del requisito.

4.1.2 Resultados del grupo focal

Los resultados del cuestionario utilizado durante el grupo focal se segmentaron en tres secciones para facilitar su análisis y fueron representados en las tablas 4.2, 4.3 y 4.4. Los resultados se tabularon y ordenaron, estableciendo las prácticas utilizadas con mayor frecuencia en la parte superior de las tablas y las utilizadas con menor frecuencia en la parte inferior.

La primera sección del cuestionario pretende identificar la frecuencia con la que se ejecutan las prácticas de gestión de requisitos por los equipos *Scrum*. Los resultados de esta sección se muestran en la Tabla 4.2, en la cual destacan las prácticas que se utilizan con más frecuencia, como lo son la identificación de la necesidad, la documentación de los requisitos en Jira y la estimación del esfuerzo. Además, destacan en las posiciones inferiores las prácticas que se realizan con menor frecuencia, como lo son la realización de prototipos, la priorización con base en el valor de negocio, la validación de que la solución satisface la necesidad, la aprobación formal de la solución y la retroalimentación por parte de los interesados en cada sprint.

Tabla 4.2. Prácticas actuales de gestión de requisitos

Práctica utilizada	OMA	TAW	KEK	IKN	Total
Se identifica la necesidad	5	3	4	4	16
Se documentan los requisitos en Jira	5	4	3	3	15
Se estima el esfuerzo	4	4	4	2	14
Se identifica a los interesados	5	3	4	1	13
Se descomponen los requisitos	4	3	3	3	13
Se clasifican y organizan los requisitos	3	4	3	3	13
Se documentan los cambios en los requisitos	2	3	3	5	13
Se obtienen los requisitos de los interesados	4	3	3	2	12
Se obtiene retroalimentación por parte del <i>Product Owner</i> en cada <i>sprint</i>	5	2	2	3	12
Se genera una línea base de requisitos	4	3	2	2	11
Se obtiene la aprobación de la línea base	4	3	3	1	11
Se comunican los cambios en los requisitos	3	3	3	2	11
Se aprueban los cambios en los requisitos	3	3	3	2	11
Se realizan prototipos	2	2	3	3	10
Se priorizan los requisitos según el valor de negocio	5	1	2	2	10
Se valida que la solución satisface la necesidad planteada	2	3	3	2	10
Se obtiene la aprobación formal de la solución por parte de los interesados	2	3	3	2	10
Se obtiene retroalimentación por parte de los interesados en cada <i>sprint</i>	3	2	2	1	8

La segunda sección del cuestionario pretende identificar la frecuencia con la que se utilizan las prácticas de formulación de requisitos. En las posiciones superiores de la Tabla 4.3 destacan la utilización de historias de usuario y las especificaciones de requisitos de desempeño del software y de base de datos. En las posiciones inferiores, se muestran la utilización del método INVEST para la redacción de las historias de usuario y la descripción de las funciones que ejecutará el software.

Tabla 4.3. Prácticas actuales de formulación de requisitos

Práctica utilizada	OMA	TAW	KEK	IKN	Total
Se utiliza la técnica de historia de usuario para la redacción	4	3	3	3	13
Se especifican los requisitos de desempeño del <i>software</i>	5	2	2	4	13
Se especifican los requisitos de base de datos	3	3	3	4	13
Se incluye una descripción clara que evita la ambigüedad	3	3	3	2	11
Se describen las entradas y salidas del sistema de <i>software</i>	2	3	3	2	10
Se especifican las restricciones de diseño	1	3	3	3	10
Se utiliza el método INVEST para la redacción de las historias de usuario	4	1	1	3	9
Se describen las funciones que ejecutará el <i>software</i>	2	3	2	1	8

Por último, la tercera sección del cuestionario pretende identificar si los equipos *Scrum* cuentan con el conocimiento y la experiencia práctica para la gestión de requisitos. En la

Tabla 4.4 destacan la identificación, estimación y documentación de requisitos como las competencias más presentes en los miembros del equipo *Scrum*. Por el contrario, las competencias de priorización y control de cambios están menos presentes.

Tabla 4.4. Competencia en gestión de requisitos

Competencia	OMA	TAW	KEK	IKN	Total
El equipo cuenta con conocimiento teórico para la identificación de los requisitos	4	3	4	5	16
El equipo cuenta con experiencia práctica para la estimación de los requisitos	4	3	4	5	16
El equipo cuenta con experiencia práctica para la identificación de los requisitos	4	2	4	5	15
El equipo cuenta con conocimiento teórico para la documentación de los requisitos	5	2	3	5	15
El equipo cuenta con conocimiento teórico para la estimación de los requisitos	2	3	4	5	14
El equipo cuenta con experiencia práctica para la documentación de los requisitos	3	2	3	5	13
El equipo cuenta con conocimiento teórico para la priorización de los requisitos	2	3	4	2	11
El equipo cuenta con experiencia práctica para el control de cambio de los requisitos	3	2	4	2	11
El equipo cuenta con experiencia práctica para la priorización de los requisitos.	1	3	4	2	10
El equipo cuenta con conocimiento teórico para el control de cambio de los requisitos	2	2	4	2	10

4.1.3 Resultados de la entrevista

Los resultados de la entrevista realizada a los *Scrum Master* de la Tribu Navi se muestran en el apéndice H. A continuación, se resumen los hallazgos de la entrevista:

- La identificación de la necesidad y la priorización de iniciativas se realiza por parte del *Chief Product Owner* en conjunto con los *Product Owners*. Estas actividades están fuera del alcance de los equipos *Scrum* ya que se realiza durante las sesiones de *MetaScrum*.
- Durante el refinamiento del *backlog* se suele realizar la identificación, documentación, estimación y control de cambios de los requisitos al criterio de cada equipo y sin un proceso definido para realizarlo.
- Durante el *sprint planning* se suele realizar la identificación, documentación y estimación de los requisitos.
- Durante del *daily Scrum* se suele atender la priorización de requisitos emergentes y el control de cambio de los requisitos, con el fin de adaptar el *sprint* al cambio.

4.1.4 Análisis de resultados

Al contrastar los resultados de la revisión documental, el grupo focal y la entrevista, es posible obtener hallazgos importantes que permiten la identificación de las prácticas realmente utilizadas por los equipos *Scrum*. A continuación, se muestran estos hallazgos:

- **Identificación de la necesidad:** Como se observa en la ficha de síntesis F-001 del apéndice F, el Manual Operativo *Scrum* especifica que el *Product Owner* es el responsable de desarrollar y comunicar explícitamente la necesidad y objetivo del producto. En la Tabla 4.2, los equipos *Scrum* establecen, en su mayoría, que casi siempre se identifica la necesidad de la solución, a pesar de esto, no es posible encontrar evidencia documentada en Jira. Por lo tanto, en coherencia con lo expuesto por los *Scrum Master* en la entrevista, la identificación de la necesidad recae en el *Chief Product Owner*, no en el equipo *Scrum*.

- **Control de cambios:** Como se observa en la ficha de síntesis F-002 del apéndice F, el *Product Owner* es el responsable de gestionar las solicitudes de incorporación, modificación y eliminación de elementos del *Product backlog*. A pesar de esto, en la Tabla 4.2, la mayoría de los equipos *Scrum* indican que el cambio en los requisitos se documenta ocasionalmente, lo cual coincide con lo expuesto en la entrevista, en la cual se asegura que los cambios no se controlan, sino que se asimilan. Es posible concluir que la práctica de control de cambios es poco utilizada debido a que no se cuenta con insumos provistos por el Manual Operativo *Scrum* y es una de las competencias menos desarrolladas según la

-

-

-

-

- **Tabla 4.4.**

- **Documentación de requisitos:** Como se observa en la ficha de síntesis F-003 del apéndice F, el Manual Operativo *Scrum* especifica el uso de historias de usuario como técnica única de redacción de los requisitos. A pesar de esto, en la Tabla 4.2 los equipos *Scrum* indicaron, en su mayoría, que la utilizan ocasionalmente. Esto queda confirmado en la Tabla 4.1, en donde se observa que los elementos de rol, expectativa y razón que conforman la historia de usuario solamente estuvieron presentes en un 44.26%, 48.89% y 22.41% respectivamente:
 - o El cumplimiento (44.26%) y la definición errónea del rol (20,74%) en las historias de usuario evidencian que no existe una identificación adecuada de los verdaderos usuarios del producto de software, lo que impide al equipo *Scrum* tener una perspectiva clara del requisito.
 - o El cumplimiento (48.89%) en la definición de la expectativa en las historias de usuario evidencian que no existe claridad durante la formulación de los requisitos, fomentando la ambigüedad y aumentando la dificultad para estimar el esfuerzo.
 - o El cumplimiento (22.41%) en la definición de la razón en las historias de usuario evidencia que los equipos *Scrum* no cuentan con claridad del objetivo de su trabajo.

- **Descomposición de requisitos:** Como se observa en la ficha de síntesis F-004 del apéndice F, el Manual Operativo *Scrum* especifica que los *developers* son responsables de disgregar los elementos del *product backlog*, pero no especifica cómo se realizará esta descomposición para facilitar la estimación del esfuerzo y evitar las dependencias.

- **Revisión de requisitos:** Como se observa en la ficha de síntesis F-005 del apéndice F, durante la finalización del *sprint*, el equipo *Scrum* se reúne con los interesados para inspeccionar los requisitos trabajados y obtener una lista de requisitos actualizada y revisada. A pesar de esto, no existe evidencia documentada de la demostración del producto desarrollado, de la retroalimentación ni de los cambios realizados a los requisitos.

- **Estimación de requisitos:** Como se observa en la ficha de síntesis F-006 del apéndice F, todos los requisitos del *product backlog* deben contar con un valor de esfuerzo estimado en puntos, utilizando las tallas XS, S, M, L y XL. En la Tabla 4.2, los equipos *Scrum*

aseguran realizar esta estimación casi siempre. Durante el análisis de la problemática, se determina que la dificultad para definir la complejidad de los requisitos es una de las principales causas del problema, ya que, durante el *sprint*, los equipos *Scrum* detectan que los requisitos son más complejos de lo que habían estimado. Por esto, es posible concluir que la práctica descrita en el Manual Operativo *Scrum* dificulta estimar con precisión, ya que indica únicamente las tallas a utilizar, pero no brinda una guía del cómo hacerlo correctamente, dejando a criterio de cada equipo *Scrum* esta tarea.

- **Identificación de requisitos:** Como se observa en la ficha de síntesis F-007 del apéndice F, al identificar los requisitos deben definirse las principales funcionalidades del producto, las historias de alto nivel, las sub-historias y los usuarios que utilizarán el producto. A pesar de esto, en la Tabla 4.3, se observa que los equipos *Scrum* ejecutan las prácticas de formulación de requisitos ocasionalmente o aún con menos frecuencia. Por lo anterior, es posible concluir que la práctica actual establece únicamente los elementos que deben obtenerse al finalizar el proceso de identificación, pero no establece el método para obtener con claridad y exactitud los requisitos directamente de los interesados y usuarios finales, fomentando así el aumento de la ambigüedad.
- **Priorización de requisitos:** Como se observa en la ficha de síntesis F-008 del apéndice F, el *Product Owner* debe priorizar constantemente adaptándose al aprendizaje generado por la retroalimentación de los interesados. En concordancia, durante la entrevista se comentó que el *Product Owner* se encarga de equilibrar las iniciativas de productos nuevos con las existentes. A pesar de esto, no existe evidencia alguna de las prácticas utilizadas por el equipo *Scrum* para la priorización. Además, en la

-
-
-
-

- **Tabla 4.4** se considera la priorización como una de las competencias menos desarrolladas en los equipos *Scrum*.

- **Procesos:** Mediante la revisión documental del Manual Operativo *Scrum* y la entrevista a los *Scrum Master* fue posible identificar prácticas aisladas de gestión de requisitos que se ejecutan durante los eventos *Scrum*, más no así, procesos claramente definidos para la gestión de requisitos. La ausencia de procesos definidos, sus entradas, herramientas, técnicas y salidas dificulta la estandarización y fomenta la informalidad. Además, no fue posible identificar una secuencia lógica de ejecución de las prácticas de gestión de requisitos ni entregables concretos.

- **Competencias:** En la
 -
 -
 -
 -

- **Tabla 4.4**, se muestra que las competencias relacionadas a la gestión de requisitos varían significativamente entre los equipos *Scrum*. Ahora bien, también es posible identificar que para cada competencia se cuenta con algún equipo *Scrum* altamente competente (4 o 5), por lo que es posible concluir que la Tribu Navi cuenta con las competencias necesarias, pero no se encuentran correctamente niveladas entre los equipos *Scrum*.

- **Estandarización:** En general, las prácticas utilizadas por los equipos *Scrum* varían significativamente. Es posible observar prácticas específicas que un equipo adopta siempre, mientras que otro lo hace nunca. Esto evidencia que no existe una estandarización de la metodología de proyectos debido principalmente a los escasos insumos proveídos por el Manual Operativo *Scrum*.

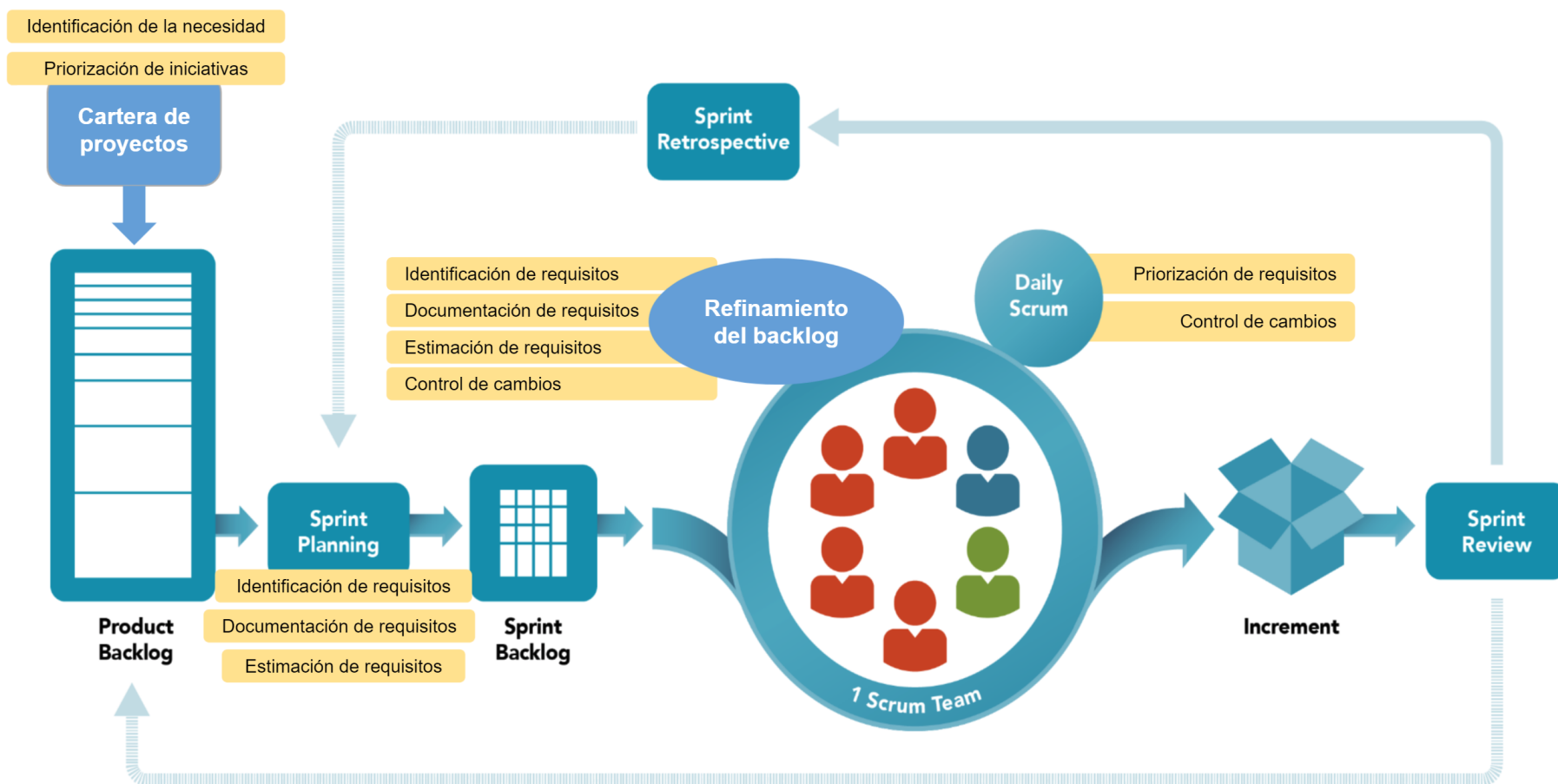
4.1.5 *Flujograma de procesos actuales de gestión de requisitos*

En el flujograma, se muestra en azul el proceso iterativo de *Scrum* con sus eventos, roles y artefactos extraídos del Manual Operativo *Scrum* y utilizados por la Tribu Navi para la gestión de sus proyectos. En amarillo, se muestran las prácticas de gestión de requisitos identificadas mediante la entrevista a los *Scrum Master*.

Figura 4.1 fue obtenido como producto de la recolección, procesamiento y análisis de los datos del Manual Operativo *Scrum* y la entrevista realizada a los *Scrum Master*. Como se explicó en el análisis de resultados, no fue posible identificar procesos documentados para la gestión de requisitos, sino más bien, una serie de prácticas aisladas de gestión de requisitos que se ejecutan durante los eventos *Scrum*.

En el flujograma, se muestra en azul el proceso iterativo de *Scrum* con sus eventos, roles y artefactos extraídos del Manual Operativo *Scrum* y utilizados por la Tribu Navi para la gestión de sus proyectos. En amarillo, se muestran las prácticas de gestión de requisitos identificadas mediante la entrevista a los *Scrum Master*.

Figura 4.1. Flujograma de procesos de gestión de requisitos



4.1.6 *Inventario de prácticas actuales de gestión de requisitos*

El inventario de prácticas actuales de gestión de requisitos fue obtenido como producto de la recolección, procesamiento y análisis de los datos. En el Cuadro 4.1., se muestra el inventario de prácticas y su relación con los elementos de la problemática expuestos en la sección 1.2.4 de análisis de causa. En la sección 4.2.4, se expondrá el análisis conjunto entre las prácticas actuales, las propuestas y los elementos de la problemática.

Cuadro 4.1. Inventario de prácticas actuales de gestión de requisitos

Inventario de prácticas actuales de gestión de requisitos								
Identificador de la práctica	Nombre de la práctica	Categoría	Elementos de la problemática a los que se relaciona o solventa					
			Complejidad	Ambigüedad	Dependencia de terceros	Dependencia entre requisitos	Cambios de prioridad	Cambios no gestionados
F-001	Identificación de la necesidad del producto	Proceso	X	X	X			
F-002	Control de cambios de requisitos	Proceso					X	X
F-003	Documentación de requisitos	Proceso	X	X		X		
F-004	Descomposición de requisitos	Proceso	X			X		
F-005	Revisión de requisitos	Proceso	X	X				X
F-006	Estimación de requisitos	Proceso	X					
F-007	Identificación de requisitos	Proceso	X	X	X	X		
F-008	Priorización de requisitos	Proceso					X	X

4.2 Buenas prácticas de gestión de requisitos

En esta sección se exponen los resultados de la revisión documental, la cual se extrajo de la literatura las buenas prácticas de gestión de requisitos. También, se exponen los resultados del análisis de casos de referencia, por medio del cual se identifican las buenas prácticas de gestión de requisitos utilizadas por organizaciones reconocidas mundialmente. Finalmente, se exponen los productos resultantes de la investigación, el inventario de buenas prácticas y el cuadro de análisis, el cual compara las actuales de los equipos *Scrum* con las identificadas.

4.2.1 Resultados de la revisión documental

La revisión documental permite extraer las buenas prácticas de gestión de requisitos, según distintos autores, que pueden ser utilizadas para solventar la problemática de la Tribu Navi. A continuación, se muestra un listado de las prácticas identificadas en el apéndice L:

- L-001 - Evaluación de la necesidad.
- L-002 - Obtención de requisitos.
- L-003 - Análisis de requisitos.
- L-004 - Monitoreo y control de requisitos.
- L-005 - Evaluación de la solución.
- L-006 - Cierre del proyecto o fase.
- L-007 - Definición de roles de usuario.
- L-008 - Obtención de historias de usuario.
- L-009 - Formulación de historias de usuario.
- L-010 - Estimación de historias de usuario.
- L-011 - Documento de requisitos del producto (PRD).
- L-012 - Hoja de ruta del producto.
- L-013 - Redacción de historias de usuario.
- L-014 - Descomposición del trabajo en Jira.
- L-015 - Estimación de historias de usuario.

4.2.2 Resultados del análisis de casos de referencia

El análisis de casos de referencia consiste en la revisión de documentos oficiales de las organizaciones analizadas para extraer las buenas prácticas de gestión de requisitos mediante fichas de síntesis. Las fichas de cada organización pueden observarse en los apéndices I, J y K.

Inicialmente, se analiza el documento *Requirements Engineering Management Handbook* del *U.S. Department of Transportation*, el cual presenta una serie de prácticas del como colectar, escribir, validar y organizar los requisitos. Además, intenta reunir las mejores ideas de varios enfoques, organizarlas coherentemente e ilustrándolas con ejemplos concretos que dejen claro su beneficio. En el apéndice I se muestran las fichas de síntesis con las buenas prácticas identificadas.

A continuación, se muestra un listado de dichas prácticas:

- I001- Desarrollar la descripción general del sistema
- I002- Identificar el alcance del sistema
- I003- Desarrollar los conceptos operacionales
- I004- Identificar los supuestos ambientales
- I005- Desarrollar la arquitectura funcional
- I006- Identificar las restricciones de implementación
- I007- Desarrollar los requisitos de comportamiento y desempeño

Se analizó también el documento *System Engineering Handbook* de la NASA, el cual brinda un enfoque de implementación de alto nivel sobre las buenas prácticas de ingeniería de sistemas utilizadas por la NASA. Este documento describe las mejores prácticas de ingeniería de sistemas que deben ser utilizadas en el desarrollo e implementación de programas y proyectos de la NASA. En el apéndice J se muestran las fichas de síntesis con las buenas prácticas identificadas.

A continuación, se muestra un listado de dichas prácticas:

- J001- Definición y gestión de las expectativas de los interesados
- J002- Definición de requisitos

- J003- Descomposición lógica
- J004- Definición del diseño de solución
- J005- Verificación del producto
- J006- Validación del producto
- J007- Planeación técnica
- J008- Gestión de requisitos
- J009- Gestión del riesgo técnico
- J010- Documentación de Requisitos
- J011- Matriz de verificación de requisitos
- J012- Estructura de desglose del producto

Por último, se analizó a la empresa Infosys, líder mundial en consultoría y servicios digitales de nueva generación con 40 años de servicio, \$14.2 billones de ingresos totales en el último año y presencia en más de 50 países. En su documento *Model-Based System Engineering* se propone un marco de trabajo que ayuda a las compañías a cambiar su gestión de requisitos basada en papel a una basada en modelos digitales.

En el apéndice K se muestran las fichas de síntesis con las buenas prácticas identificadas.

A continuación, se muestra un listado de dichas prácticas:

- K001- Documentación de requisitos
- K002- Modelado de requisitos
- K003- Matriz de requisitos
- K004- Descomposición de requisitos
- K005- Diagrama de estructura
- K006- Diagrama de comportamiento
- K007- Diagrama paramétrico

4.2.3 *Inventario de buenas prácticas de gestión de requisitos*

En el

Cuadro 4.2, se muestra el inventario de buenas prácticas de gestión de requisitos. Estas prácticas compuestas de procesos, técnicas y herramientas conforman una base de conocimiento suficientemente robusta para sustentar la propuesta de solución. Además, se detalla cada uno de los elementos de la problemática que solventan las prácticas identificadas.

Cuadro 4.2. Inventario de buenas prácticas de gestión de requisitos

Inventario de buenas prácticas de gestión de requisitos								
Identificador de la práctica	Nombre de la práctica	Categoría	Elementos de la problemática a los que se relaciona o solventa					
			Complejidad	Ambigüedad	Dependencia de terceros	Dependencia entre requisitos	Cambios de prioridad	Cambios no gestionados
I-001	Desarrollar la descripción general del sistema	Proceso	X	X				
I-002	Identificar el alcance del sistema	Proceso	X	X				
I-003	Desarrollar los conceptos operacionales	Proceso	X	X		X		
I-004	Identificar los supuestos ambientales	Proceso	X	X				
I-005	Desarrollar la arquitectura funcional	Proceso				X		
I-006	Identificar las restricciones de implementación	Proceso	X	X		X		
I-007	Desarrollar los requisitos de comportamiento y desempeño	Proceso	X	X				
J-001	Definición y gestión de las expectativas de los interesados	Proceso	X	X	X		X	X
J-002	Definición de requisitos	Proceso	X	X		X		

J-003	Descomposición lógica	Proceso	X	X		X		
J-004	Definición del diseño de solución	Proceso	X	X	X	X		

Cuadro 4.2. Inventario de buenas prácticas de gestión de requisitos (continuación)

Inventario de buenas prácticas de gestión de requisitos								
Identificador de la práctica	Nombre de la práctica	Categoría	Elementos de la problemática a los que se relaciona o solventa					
			Complejidad	Ambigüedad	Dependencia de terceros	Dependencia entre requisitos	Cambios de prioridad	Cambios no gestionados
J-005	Verificación del producto	Proceso			X		X	
J-006	Validación del producto	Proceso			X		X	
J-007	Planeación técnica	Proceso			X		X	
J-008	Gestión de requisitos	Proceso	X	X	X	X	X	X
J-009	Gestión del riesgo técnico	Proceso					X	X
J-010	Documentación de requisitos	Técnica	X	X				
J-011	Matriz de verificación de requisitos	Herramienta	X	X				
J-012	Estructura de desglose del producto	Herramienta	X	X	X	X		
K-001	Documentación de requisitos	Técnica		X				
K-002	Modelado de requisitos	Técnica	X	X		X		X
K-003	Matriz de requisitos	Herramienta				X		X
K-004	Descomposición de requisitos	Técnica	X			X		
K-005	Diagrama de estructura	Herramienta	X	X		X		

K-006	Diagrama de comportamiento	Herramienta	X	X		X		
K-007	Diagrama paramétrico	Herramienta	X	X		X		

Cuadro 4.2. Inventario de buenas prácticas de gestión de requisitos (continuación)

Inventario de buenas prácticas de gestión de requisitos								
Identificador de la práctica	Nombre de la práctica	Categoría	Elementos de la problemática a los que se relaciona o solventa					
			Complejidad	Ambigüedad	Dependencia de terceros	Dependencia entre requisitos	Cambios de prioridad	Cambios no gestionados
L-001	Evaluación de la necesidad	Proceso	X	X	X			
L-002	Obtención de requisitos	Proceso	X	X	X	X		
L-003	Análisis de requisitos	Proceso	X	X	X	X		
L-004	Monitoreo y control de requisitos	Proceso			X	X	X	X
L-005	Evaluación de la solución	Proceso	X	X			X	X
L-006	Cierre del proyecto o fase	Proceso					X	X
L-007	Definición de roles de usuario	Proceso	X	X	X			
L-008	Obtención de historias de usuario	Proceso	X	X	X	X		
L-009	Formulación de historias de usuario	Técnica	X	X	X	X		
L-010	Estimación de historias de usuario	Técnica	X					
L-011	Documento de requisitos del producto (PRD)	Herramienta	X	X	X	X	X	X
L-012	Hoja de ruta del producto	Herramienta			X	X	X	X
L-013	Redacción de historias de usuario	Técnica	X	X		X		
L-014	Descomposición del trabajo en Jira	Técnica	X			X		
L-015	Estimación de historias de usuario	Técnica	X					

4.2.4 Cuadro de análisis

El Cuadro 4.3 presenta una comparativa de las prácticas actuales y las buenas prácticas seleccionadas, en busca de identificar alternativas de mejora de los procesos, técnicas y herramientas actuales. Las prácticas se categorizaron con base en los procesos de gestión de requisitos establecidos en la sección 2.2.1 del marco teórico y que a su vez forman parte de las buenas prácticas identificadas. El análisis detalla la razón por la cual cada práctica ha sido seleccionada para formar parte de la propuesta de solución a la problemática.

Cuadro 4.3. Cuadro de análisis de prácticas actuales y buenas prácticas seleccionadas

Cuadro de análisis de prácticas actuales y buenas prácticas seleccionadas		
Evaluación de la necesidad		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
F-001 Identificación de la necesidad del producto	J-001 Definición y gestión de las expectativas de los interesados	<ul style="list-style-type: none"> - Debido a que la práctica actual de identificación de la necesidad recae en el <i>Chief Product Owner</i>, el equipo <i>Scrum</i> no cuenta con contacto directo con los interesados ni sus expectativas, lo que aumenta la dificultad de definir la complejidad de la solución, fomenta la ambigüedad y genera dependencias de terceras personas. - La implementación de la práctica J-001, permitirá identificar la necesidad y expectativas directamente de los interesados y mantener contacto con ellos durante el desarrollo de los requisitos.
Planificación de la gestión de requisitos		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
No se identificaron prácticas relacionadas.	J-008 Gestión de requisitos L-011 Documento de requisitos del producto (PRD)	<ul style="list-style-type: none"> - Debido a que no fueron identificadas prácticas actuales de planificación de la gestión de requisitos, la implementación de la práctica J-008 como proceso para la elaboración del plan de gestión y la práctica L-011 como herramienta de gestión de requisitos en Jira permitirán a los equipos <i>Scrum</i> formalizar sus procesos, técnicas y herramientas utilizadas durante la ejecución del plan de gestión de requisitos.

Cuadro 4.3. Cuadro de análisis de prácticas actuales y buenas prácticas (continuación)

Cuadro de análisis de prácticas actuales y buenas prácticas seleccionadas		
Obtención de requisitos		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
F-003 Documentación de requisitos	L-007 Definición de roles de usuario	<ul style="list-style-type: none"> - Las prácticas actuales F-003 y F-007 establecen que los requisitos deben documentarse en Jira mediante la técnica de historia de usuario en cumplimiento con el método INVEST y además deben incluir las funcionalidades, las sub-historias y la identificación de los usuarios. A pesar de esto, las prácticas no especifican cómo se realizará la obtención de los requisitos ni la identificación de los roles de usuario, ya que únicamente abarca cómo serán documentados una vez obtenidos. - Las prácticas L-007 y L-008 brindan insumos para la correcta definición de los roles de usuario y las historias de usuario de forma que logren definir con mayor precisión los requisitos. - La práctica L-009 brinda recomendaciones útiles al formular las historias de usuario y la práctica J-010 incluye una lista de verificación y validación de la forma en la que están redactados los requisitos.
F-007 Identificación de requisitos	L-008 Obtención de historias de usuario	
	L-009 Formulación de historias de usuario	
	J-010 Documentación de requisitos	
Análisis de requisitos		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
F-004 Descomposición de requisitos	J-003 Descomposición lógica	<ul style="list-style-type: none"> - Las prácticas actuales F-004, F-006 y F-008 establecen que las historias de alto nivel deben ser descompuesta y priorizadas de manera que sea posible entregar valor en cada <i>sprint</i>. Además, establecen que todos los requisitos del <i>Product backlog</i> deben contar con un valor de esfuerzo estimado en puntos con base en las tallas XS, S, M, L y XL. Ahora bien, estas prácticas no ofrecen técnicas ni herramientas para su correcta ejecución, provocando dificultad para definir la complejidad de cada requisito, elevando las dependencias entre sí y fomentando los cambios de prioridad constantes. - La práctica J-003 detalla el proceso de descomposición de los requisitos, sus entradas, subprocesos y salidas. La práctica J-012 es una herramienta que permite representar gráficamente la descomposición de los requisitos, ayuda a identificar su nivel de complejidad, las dependencias entre sí y facilita la comprensión de la línea base de requisitos. - La práctica L-010 define las técnicas y herramientas a utilizar para la correcta estimación mediante puntos de historia. - La práctica L-003 establece técnicas de gestión y priorización del <i>backlog</i> que permiten establecer con certeza la prioridad de cada requisito y un ordenamiento lógico de desarrollo.
F-006 Estimación de requisitos	J-012 Estructura de desglose del producto	
F-008 Priorización de requisitos	L-010 Estimación de historias de usuario	
	L-003 Análisis de requisitos	

Cuadro 4.3. Cuadro de análisis de prácticas actuales y buenas prácticas (continuación)

Cuadro de análisis de prácticas actuales y buenas prácticas seleccionadas		
Evaluación de la solución		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
F-005 Revisión de requisitos	J-005 Verificación del producto	<ul style="list-style-type: none"> - La práctica actual F-005 define con claridad el objetivo de la sesión de revisión de requisitos y sus participantes, pero no define los entregables esperados de la sesión, ni cómo serán construidos. - Las prácticas J-005 y J006 detallan los procesos de verificación y validación del producto en busca de comprobar que el producto satisfice los requisitos y las expectativas de los interesados. - La práctica L-005 aporta técnicas para la evaluación de la solución que fortalecen los procesos de verificación y validación del producto, como solicitar opinión de los usuarios, ejecutar pruebas o realizar demostraciones.
	J-006 Validación del producto	
	L-005 Evaluación de la solución	
Monitoreo y control de requisitos		
Prácticas actuales	Buenas prácticas seleccionadas	Análisis de la problemática y su solución
F-002 Control de cambios de requisitos	L-004 Monitoreo y control de requisitos	<ul style="list-style-type: none"> - La práctica actual F-002 establece que el <i>Product Owner</i> es el responsable de gestionar el cambio en los requisitos, pero no especifica cómo o en qué momento lo realizará, ni define el cómo se involucrará al resto del equipo <i>Scrum</i> para lograrlo. - La práctica L-004 define la utilización de la matriz de requisitos para optimizar el monitoreo y control de los requisitos desde su creación hasta el entregable que lo satisface.

Capítulo 5. Propuesta de Solución

En este capítulo se desarrolla la propuesta de solución de la problemática, la cual fue construida a partir de los resultados obtenidos en la investigación y fue desarrollada para satisfacer los objetivos planteados y sus entregables correspondientes. La solución consiste en un método estándar de gestión de requisitos de proyectos para la Tribu Navi. Este método está conformado de un flujograma de procesos, una guía metodológica y un diccionario de términos para la gestión de requisitos.

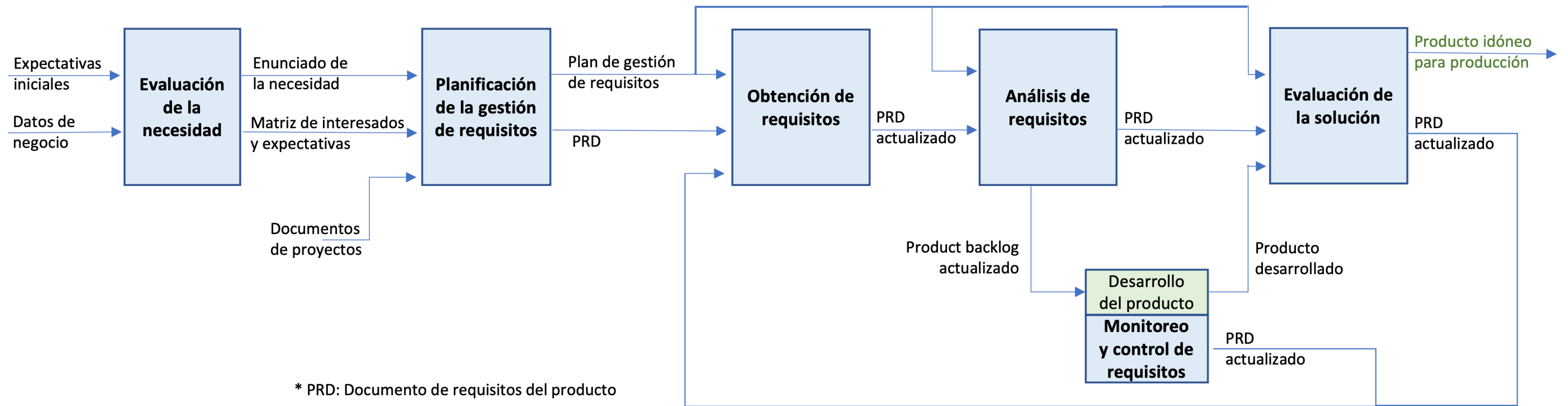
5.1 Flujograma de procesos de gestión de requisitos

El flujograma general de procesos de gestión de requisitos se muestra en la Figura 5.1 y forma parte de la guía metodológica propuesta en la sección 5.2, en la cual se puede encontrar la descripción de cada uno de los procesos. Este flujograma de bloques muestra las entradas y salidas de cada proceso, así como el ciclo iterativo de obtención de requisitos, análisis de requisitos y evaluación de la solución. Este ciclo iterativo parte del hecho de que al inicio del proyecto los requisitos no son conocidos en su totalidad y que conforme se ejecutan las iteraciones es posible identificar nuevos requisitos, mejorar su definición y adecuar el producto para satisfacer las expectativas de los interesados.

Los procesos propuestos deberán ser ejecutados conforme a lo establecido en la guía metodológica y estarán integrados a los eventos *Scrum* según lo establezca el mismo equipo *Scrum* en el plan de gestión de requisitos desarrollado para el proyecto específico.

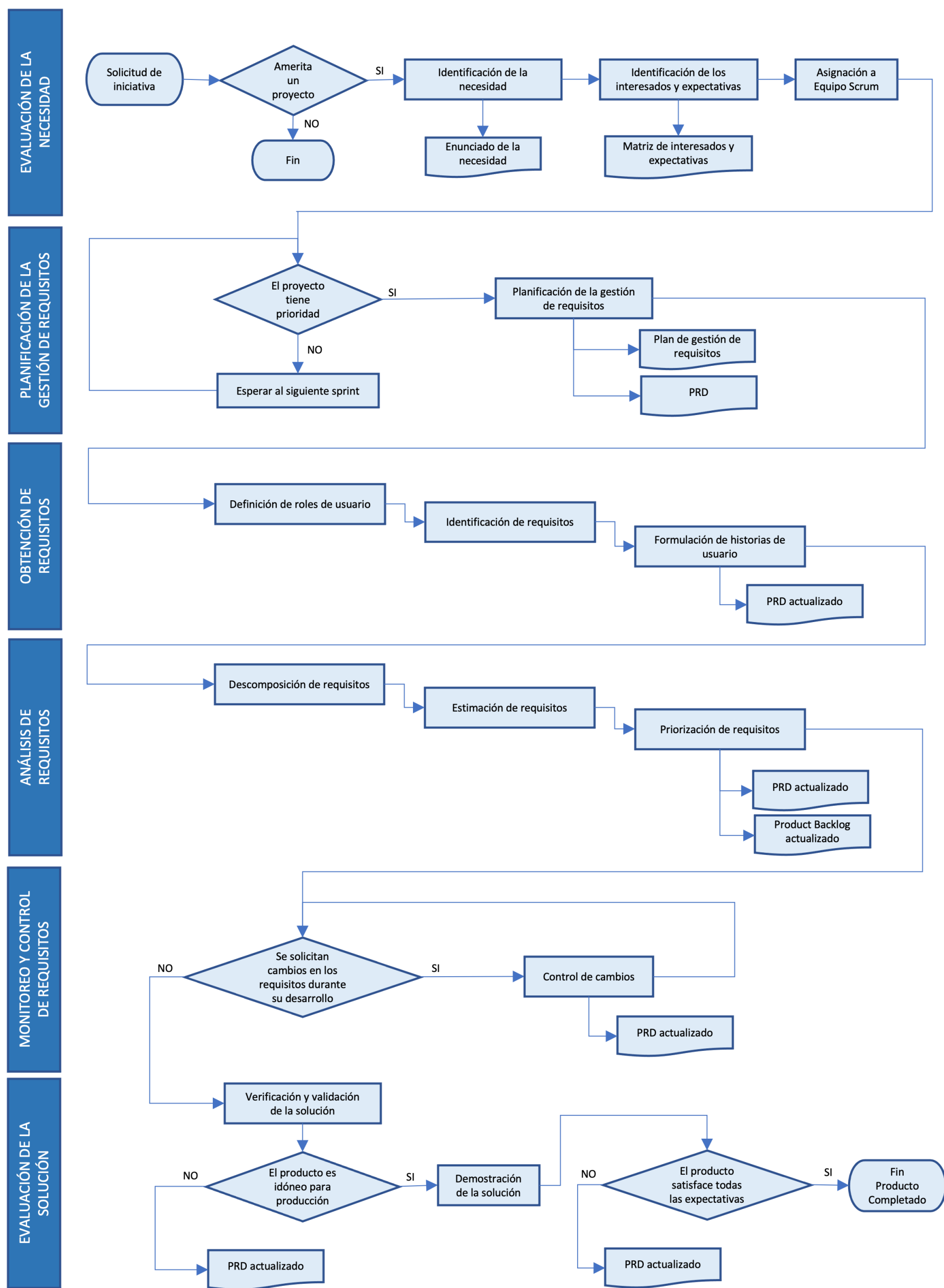
El flujograma muestra en color azul los nuevos procesos propuestos para la gestión de requisitos y en color verde el proceso de desarrollo de software, el cual no forma parte de las técnicas de gestión de requisitos, al igual que la salida final, la cual consiste en el producto idóneo para producción.

Figura 5.1. Flujograma general de procesos de la gestión de requisitos.



El flujograma detallado de procesos de gestión de requisitos se muestra en la Figura 5.2. Este flujograma detalla cada uno de los procesos, actividades, decisiones y productos.

Figura 5.2. Flujograma detallado de procesos de la gestión de requisitos



5.2 Guía metodológica para la gestión de requisitos

En esta sección se exponen los elementos de la guía metodológica propuesta como solución a la problemática identificada en la Tribu Navi.

5.2.1 Introducción

Esta guía metodológica tiene como objetivo documentar los procesos, técnicas y herramientas que permitan la adecuada gestión de requisitos en la Tribu Navi. Una vez sea implementada y adoptada por los equipos de trabajo, se espera alcanzar una estandarización de las prácticas ejecutadas en concordancia con los principios ágiles establecidos en el marco de trabajo *Scrum*.

La guía metodológica consiste en la sistematización y documentación de los procesos, técnicas y herramientas necesarios para la gestión de requisitos de proyectos en la Tribu Navi. Se compone de secciones que definen las actividades a realizar para obtener los productos resultantes de cada uno de los procesos.

5.2.2 Proceso de evaluación de la necesidad

El proceso de evaluación de la necesidad, mostrado en la **Figura 5.3**, tiene como objetivo definir el problema u oportunidad de negocio que el proyecto pretende atender, con el fin de brindar al equipo *Scrum* una perspectiva que le permita optar por una o varias alternativas de solución que brinden un verdadero valor de negocio.

Figura 5.3. Proceso de evaluación de la necesidad



5.2.2.1 Entrada: Expectativas iniciales de los interesados

El conjunto de expectativas iniciales de los interesados consiste en un listado verbal o escrito de las aspiraciones del promotor, patrocinador y demás interesados con respecto a la solución del problema u oportunidad de negocio que se presenta. Esta lista de expectativas iniciales debe reflejar la importancia y urgencia de la solución desde la perspectiva de negocio.

5.2.2.2 Entrada: Datos de negocio

Los datos de negocio se componen de todos aquellos documentos que permiten identificar los factores que rodean la problemática u oportunidad de negocio. Entre ellos, estados financieros, reportes de negocio, análisis de brechas, lineamientos y regulaciones.

5.2.2.3 Técnica y herramienta: Sesión de identificación de la necesidad

Esta sesión tiene como objetivo identificar y limitar la necesidad de negocio. Durante este taller se evaluará si las expectativas iniciales de los interesados y los datos de negocio provistos representan una necesidad de negocio que amerite la creación de un nuevo proyecto. De ser así, se evaluará la importancia y urgencia de dicha necesidad

con el fin de determinar si el nuevo proyecto debe ser asignado a un equipo de trabajo o será colocado en la cola de la cartera de proyectos.

La identificación de la necesidad será realizada durante la sesión de *MetaScrum* y participarán en ella el promotor de la iniciativa, el *Chief Product Owner*, los *Product Owner* de cada equipo *Scrum* y los invitados que se consideren necesarios para alcanzar el objetivo.

5.2.2.4 Técnica y herramienta: Sesión de identificación de los interesados

Esta sesión tiene como objetivo identificar a los interesados y sus expectativas de la solución de la problemática u oportunidad de negocio. Debe considerarse como un interesado a todo aquel individuo, grupo u organización que tiene una relación directa o indirecta con la solución, durante y después de su desarrollo.

La identificación de los interesados será realizada durante la sesión de *MetaScrum* y participarán en ella el promotor de la iniciativa, el *Chief Product Owner*, los *Product Owner* de cada equipo *Scrum* y los invitados que se consideren necesarios para alcanzar el objetivo.

5.2.2.5 Salida: Enunciado de la necesidad

El enunciado de la necesidad consiste en una descripción concreta de la necesidad que se desea satisfacer desde el punto de vista del negocio. Este enunciado debe exponer con claridad el problema u oportunidad de negocio que se presenta, la importancia y urgencia de su solución y la relación de la solución con los interesados clave. El enunciado no debe limitar las alternativas de solución, por lo que debe evitar involucrar aspectos técnicos o de diseño. En la

Figura 5.4, se muestra la plantilla del enunciado de la necesidad, el cual será un componente del documento de requisitos del producto (PRD).

Figura 5.4. Plantilla del enunciado de la necesidad

Enunciado de la Necesidad

Descripción concreta de la necesidad que se desea satisfacer desde el punto de vista del negocio.

5.2.2.6 Salida: Matriz de interesados y expectativas

La matriz de interesados y expectativas consiste en un cuadro que relaciona a los interesados con sus expectativas específicas de la solución. Además, incluye información de contacto de cada uno de los interesados que facilitará la comunicación. En la Figura 5.5, se muestra la plantilla de matriz de interesados y expectativas, la cual será un componente del documento de requisitos del producto (PRD).

Figura 5.5. Plantilla de la matriz de interesados y expectativas

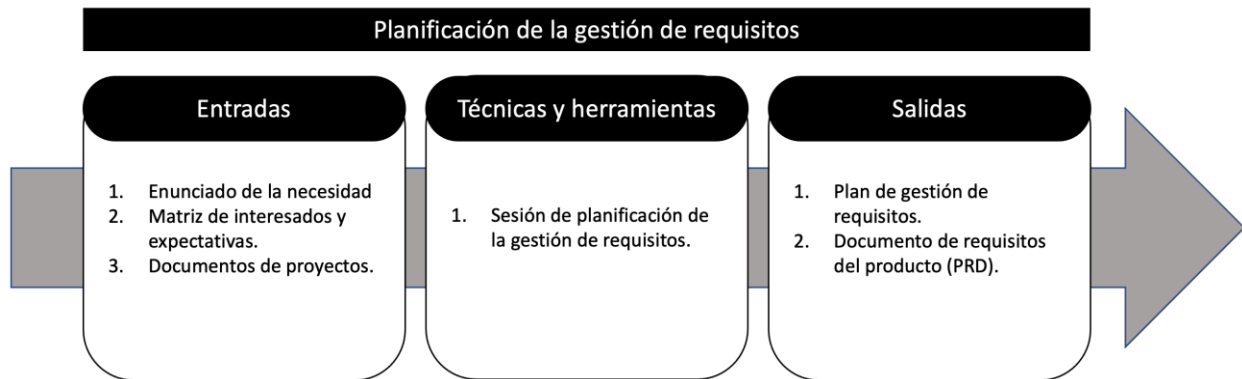
Matriz de interesados y expectativas

Nombre del Interesado	Rol del interesado	Datos de contacto	Expectativas
			<input type="checkbox"/> Expectativa 1 <input type="checkbox"/> Expectativa 2
			<input type="checkbox"/> Expectativa 1 <input type="checkbox"/> Expectativa 2

5.2.3 Proceso de planificación de la gestión de requisitos

El proceso de planificación de la gestión de requisitos, mostrado en la Figura 5.6, tiene como objetivo establecer todas las actividades que serán llevadas a cabo para la gestión de los requisitos del proyecto específico, con el fin de lograr una integración de estas actividades con las previamente definidas dentro del marco de trabajo *Scrum*.

Figura 5.6. Proceso de planificación de la gestión de requisitos



5.2.3.1 Entrada: Enunciado de la necesidad

El enunciado de la necesidad es expuesto en la sección 5.2.2.5.

5.2.3.1 Entrada: Matriz de interesados y expectativas

La matriz de interesados y expectativas es expuesta en la sección 5.2.2.6.

5.2.3.2 Entrada: Documentos de proyectos

Los documentos de proyectos consisten en todos aquellos escritos que brindan una guía del proceso de planificación de la gestión de requisitos con base en la experiencia adquirida de proyectos similares ejecutados en el pasado o que sugieren una buena práctica. Ejemplos de estos documentos son: planes de gestión de proyectos similares, registros de lecciones aprendidas, registros de sesiones de retrospectiva,

retroalimentación de los interesados, guías de procedimientos, manuales operativos y lineamientos corporativos.

5.2.3.3 Técnica y herramienta: Sesión de planificación de la gestión de requisitos

La sesión de planificación de la gestión de requisitos tiene como objetivo definir las actividades iterativas a ser ejecutadas para la gestión de requisitos del proyecto durante los *sprints*, así como sus responsables y los eventos *Scrum* relacionados. Esta sesión será el primer contacto del equipo *Scrum* con el nuevo proyecto, por lo que es de suma importancia que sea ejecutada de forma previa al inicio del desarrollo de la solución.

Esta sesión será convocada por el *Product Owner* cuando le sea asignado un nuevo proyecto con prioridad para ser ejecutado de inmediato. Si el proyecto asignado no cuenta con prioridad para su inicio, el *Product Owner* no convocará la sesión y se mantendrá a la espera de prioridad, esto con la finalidad de mantener el enfoque en el equipo *Scrum*. La sesión es considerada parte del refinamiento del *backlog* y en ella participarán los miembros del equipo *Scrum* y los invitados que se consideren necesarios para alcanzar su objetivo.

Durante la sesión, el *Product Owner* expondrá el enunciado de la necesidad, la lista de interesados y expectativas y los documentos de proyectos, con la finalidad de brindar claridad sobre el alcance, la importancia y la urgencia de la solución requerida. En conjunto, el equipo *Scrum* discutirá y definirá la mejor manera de gestionar los requisitos del proyecto, con base en lo expuesto por el *Product Owner*, su experiencia previa y esta guía metodológica. Al finalizar la sesión, se obtendrán el plan de gestión de requisitos del proyecto y la primera versión del documento de requisitos del

producto (PRD). Esta primera versión del PRD incluirá los datos del proyecto, el enunciado de la necesidad y la lista de interesados y expectativas.

5.2.3.4 Salida: Plan de gestión de requisitos

El plan de gestión de requisitos consiste en un documento que detalla las actividades de gestión de requisitos a ser ejecutadas para el proyecto específico. El plan debe incluir las actividades, cuándo y dónde serán realizadas, su facilitador y sus participantes. Las actividades contenidas en el plan de gestión son todas aquellas relacionadas a la obtención, análisis de los requisitos y evaluación de la solución que se ejecutan de manera iterativa en cada *sprint* mientras perdure la ejecución del proyecto, además de, las actividades asociadas al monitoreo y control de los requisitos.

El plan de gestión debe ser comunicado formalmente y ser visible para todos los interesados participantes en las actividades, incluyendo clientes, usuarios finales, proveedores, especialistas técnicos, jefaturas y otros equipos *Scrum*. Se destinará una página web de Confluence para alojar el plan de gestión de los requisitos del proyecto.

En el apéndice M se muestra la plantilla del plan de gestión de los requisitos.

5.2.3.5 Salida: Documento de requisitos del producto (PRD)

El PRD consiste en un documento de una sola página que define los requisitos del producto. El PRD unifica la documentación de los requisitos del producto en una única página web de Confluence, lo que facilita su comprensión, análisis y modificación por parte del equipo *Scrum* y demás especialistas técnicos. Además, simplifica la visualización y aprobación por parte de los interesados al poder ser consultado vía web y registrar comentarios de aceptación. También, mantiene un historial de cambios del documento que facilita su trazabilidad. Por último, permite la

integración con Jira, de forma que es posible conocer el estado de cada requisito en tiempo real.

El PRD se compone de los siguientes elementos, los cuales son creados y actualizados durante las actividades definidas en el plan de gestión de requisitos:

- Datos del proyecto: nombre del proyecto, Product Owner y equipo Scrum asignados.
- Enunciado de la necesidad
- Matriz de interesados y expectativas
- Matriz de trazabilidad de requisitos
- Material documental o gráfico de apoyo
- Preguntas generadoras
- Registro de cambios

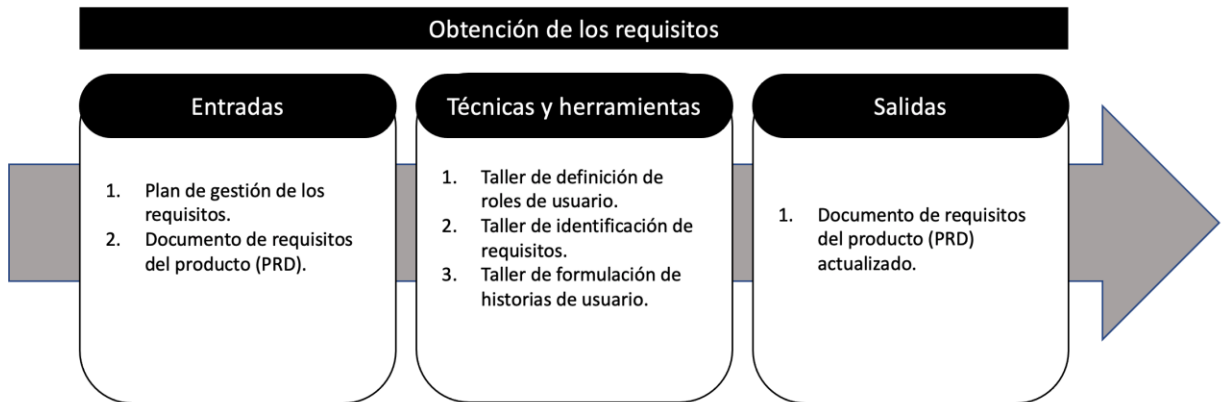
En el apéndice N se muestra la plantilla del documento de requisitos de producto.

5.2.4 Proceso de obtención de los requisitos

El proceso iterativo de obtención de los requisitos, mostrado en la Figura 5.7, tiene como objetivo extraer la información relevante que permita identificar con claridad y certeza los requisitos del producto. Estos deben ser redactados de manera que expresen la funcionalidad desde la perspectiva del usuario y ser integrados al documento de requisitos del producto (PRD) actualizado.

Debido a que inicialmente no se conocen con certeza todos los requisitos del producto, la obtención de los mismos deberá ser ejecutada de forma iterativa, al igual que los procesos de análisis de los requisitos y evaluación de la solución. Conforme se ejecuten las iteraciones y se reciba retroalimentación de los usuarios, será posible mejorar la especificación de requisitos, acercando el producto a la verdadera solución requerida.

Figura 5.7. Proceso de obtención de los requisitos



5.2.4.1 Entrada: Plan de gestión de los requisitos

El plan de gestión de los requisitos es expuesto en la sección 5.2.3.4.

5.2.4.2 Entrada: Documento de requisitos del producto (PRD)

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.4.3 Técnica y herramienta: Taller de definición de roles de usuario

El taller de definición de roles de usuario tiene como objetivo identificar con certeza cada uno de los roles de usuario que interactuarán con el producto, con el fin de permitir una correcta obtención de los requisitos al extraer información directamente del usuario y formular las historias de usuario desde su perspectiva.

El taller consiste en una sesión de trabajo conjunto entre el equipo *Scrum*, los interesados clave y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante el taller, los participantes proponen nombres para los roles de usuario y los colocan en una pizarra visible para todos, cada uno de estos roles debe representar a un individuo que interactúa con el producto, no a un colectivo. Posteriormente, los participantes agrupan los roles relacionados y eliminan aquellos que se consideran

repetidos o que no son relevantes para el éxito de la solución. Finalmente, se definen atributos para cada rol de usuario que faciliten su identificación y los diferencie del resto. Al finalizar el taller, se obtiene un listado de roles de usuario que servirá como base para el taller de identificación de requisitos.

5.2.4.4 Técnica y herramienta: Taller de identificación de requisitos

El taller de identificación de requisitos tiene como objetivo identificar los requisitos de alto nivel desde la perspectiva de cada uno de los roles de usuario, con el fin de establecer con claridad las expectativas de cada usuario con relación al producto.

El taller consiste en una sesión de trabajo conjunto entre el equipo *Scrum*, los interesados clave y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante el taller, los participantes eligen el rol de usuario más relevante y escriben en una pizarra todas las funcionalidades del producto que consideran necesarias para dicho rol. Posteriormente, los participantes ordenan las funcionalidades según su importancia y eliminan las que se consideran repetidas. Finalmente, los miembros del equipo *Scrum* dibujan un prototipo de baja fidelidad que muestre los componentes del producto y sus funcionalidades para el rol de usuario elegido. Una vez, todos los participantes estén de acuerdo con el prototipo del producto para el rol seleccionado, se repite el procedimiento para cada rol de usuario. Los prototipos generados durante el taller deberán ser adjuntados al PRD.

5.2.4.5 Técnica y herramienta: Taller de formulación de historias de usuario

El taller de formulación de historias de usuario tiene como objetivo formular los requisitos de alto nivel mediante la técnica de historia de usuario desde la

perspectiva de cada uno de los roles de usuario, con el fin de representar fielmente su expectativa del producto.

El taller consiste en una sesión de trabajo conjunto entre el equipo *Scrum*, los interesados clave y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante el taller, los participantes eligen el rol de usuario más relevante y escriben en una pizarra todas las historias relacionadas a las funcionalidades del producto representadas en el prototipo de baja fidelidad. Además, serán adjuntados a cada una de ellas, todos los criterios de aceptación del producto que consideren relevantes y el material documental o gráfico de apoyo con el que se cuente. Una vez finalizado esto, se repite el procedimiento con todos los roles de usuario.

La formulación de las historias de usuario debe respetar los siguientes lineamientos:

- Utilizar el formato: "Como" + **rol de usuario** + "requiero" + **expectativa** + "para" + **razón**.
- Definir siempre un rol de usuario válido, uno que interactúe con el producto.
- Definir una expectativa única que refleje la funcionalidad requerida de forma concreta.
- Definir una razón que permita comprender el para qué se requiere la funcionalidad.
- Si un requisito no puede ser representado desde la perspectiva del usuario, no se debe utilizar la técnica de historia de usuario para su formulación.

Una vez fueron redactadas todas las historias de usuario, se deben realizar las siguientes validaciones:

- La historia de usuario está gramaticalmente correcta.

- La historia de usuario está libre de errores tipográficos, ortográficos y de puntuación.
- La historia de usuario está redactada en positivo.
- La historia de usuario expresa una única sentencia.
- La historia de usuario está libre de pronombres indefinidos (este, estos).
- La historia de usuario está libre de términos ambiguos ("según sea apropiado", "etc.", "y/o").
- La historia de usuario está libre de términos no verificables (flexible, fácil, suficiente, seguro, amigable, rápido, portable, pequeño, grande, liviano, robusto o suficiente).
- Las historias de usuario son consistentes y no presentan contradicciones.
- Las historias de usuario utilizan terminología consistente con la terminología del usuario.
- Las historias de usuario son independientes entre sí.
- Las historias de usuario incluyen la detección, reporte, manejo y recuperación de errores.
- Las historias de usuario pueden ser verificadas mediante casos de prueba o demostraciones.

Al finalizar el taller, las historias de usuario que han superado las validaciones pasan a formar parte de la matriz de trazabilidad de requisitos del PRD, en la columna de requisitos de primer nivel. Ver apéndice N.

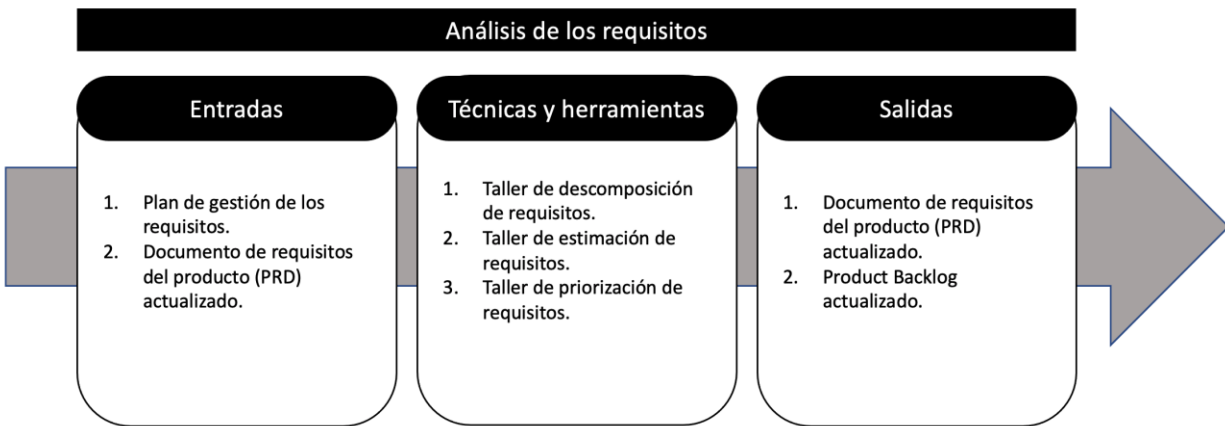
5.2.4.6 Salida: Documento de requisitos del producto (PRD) actualizado

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.5 Proceso de análisis de los requisitos

El proceso iterativo de análisis de los requisitos, mostrado en la Figura 5.8, tiene como objetivo examinar, descomponer, estimar y priorizar los requisitos formulados, con el fin de obtener un conjunto accionable de requisitos que satisfaga las expectativas planteadas.

Figura 5.8. Proceso de análisis de los requisitos



5.2.5.1 Entrada: Plan de gestión de los requisitos

El plan de gestión de los requisitos es expuesto en la sección 5.2.3.4.

5.2.5.2 Entrada: Documento de requisitos del producto (PRD) actualizado

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.5.3 Técnica y herramienta: Taller de descomposición de requisitos

El taller de descomposición de requisitos tiene como objetivo descomponer los requisitos de alto nivel en obligaciones puntuales, únicas e independientes, con el fin de disminuir su complejidad y ambigüedad al definir con mayor exactitud las expectativas y los criterios de aceptación.

El taller consiste en una sesión de trabajo conjunto entre el equipo *Scrum* y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante el taller, los participantes eligen una de las historias de usuario de alto nivel definidas en el PRD. Posteriormente, se examina la historia de usuario en busca de determinar si es posible descomponerla en sub-historias de segundo nivel más simples e independientes entre sí. A su vez, estas serán examinadas para determinar si es posible descomponerlas en un tercer nivel.

Si las sub-historias fuesen dependientes entre sí o consisten en actividades técnicas del proceso de desarrollo, deberán ser descartadas y mantener la historia de usuario original sin descomponer. Una vez haya sido descompuesta, se repite el procedimiento con el resto y se incluyen en la matriz de trazabilidad de requisitos del PRD, según su nivel correspondiente. Ver apéndice N.

5.2.5.4 Técnica y herramienta: Taller de estimación de requisitos

El taller de estimación de requisitos tiene como objetivo definir el esfuerzo en puntos que será necesario para desarrollar dicho requisito, con el fin de brindar un estimado de la cantidad de iteraciones que serán necesarias para entregar un conjunto específico de requisitos.

El taller consiste en una sesión de trabajo exclusiva del equipo *Scrum*, aunque también podrá participar en ella personal externo, siempre y cuando sean parte del equipo que desarrolla la solución. Durante el taller, los participantes seleccionan una historia de usuario, se leerá en voz alta incluyendo sus criterios de aceptación y se mostrará cualquier material documental o gráfico de apoyo que aclare el alcance de la historia. Posteriormente, los participantes realizarán preguntas o harán comentarios si

creen necesario aclarar algún aspecto del esfuerzo que implica la elaboración de la solución. Seguidamente, cada participante involucrado en el proceso de desarrollo procede a estimar la historia con base en lo siguiente:

- La estimación considera el esfuerzo grupal, no individual. Esto quiere decir que se estima el esfuerzo de todo el equipo *Scrum* enfocado en completar la historia de usuario.
- Al estimar se debe considerar el grado de complejidad, ambigüedad e incertidumbre del requisito para evitar subestimaciones.
- Los valores de estimación son XS, S, M, L o XL.
- Se utiliza como referencia el concepto de "día ideal". Un "día ideal" es aquel en el cual todos los miembros del equipo *Scrum* estuvieron enfocados en el desarrollo sin interrupciones.
- Una historia de usuario que es posible desarrollar por completo en un "día ideal" se estima con un valor de S. El resto de las historias se estiman en proporción a esta.

Una vez que todos los participantes hayan escrito su estimado, lo mostrarán al resto. Si las estimaciones coinciden, se asignará la estimación definida a la historia de usuario en la matriz de requisitos del PRD. Si existieran discrepancias, los participantes debatirán hasta alcanzar un consenso que les permita definir la estimación correcta a la historia de usuario. Finalmente, el proceso se repite para cada historia de usuario que requiera ser estimada.

5.2.5.5 Técnica y herramienta: Taller de priorización de requisitos

El taller de priorización de requisitos tiene como objetivo categorizar los requisitos según su aporte a la satisfacción de las expectativas de los interesados, con el fin de generar el mayor valor de negocio en la menor cantidad de iteraciones posible.

La priorización permitirá definir el orden de desarrollo de la solución para satisfacer los requisitos más determinantes para su éxito, con base en la asignación de uno de los siguientes atributos de priorización:

- Imprescindible: requisito que es fundamental para el éxito de la solución.
- Importante: requisito que fortalece la solución, pero que, de no realizarse no compromete el éxito de la solución.
- Deseable: requisito que puede ser postergado a futuras iteraciones sin ningún impacto.
- Prescindible: requisito que no será trabajado en esta iteración.

El taller consiste en una sesión de trabajo entre el equipo *Scrum* y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante la sesión, los participantes seleccionarán una historia de usuario y discutirán el impacto que esta tendrá con respecto a los requisitos de alto nivel y a las expectativas de los interesados. Una vez se haya alcanzado un consenso, se asignará el atributo de prioridad a dicha historia en el PRD. Finalmente, el proceso se repite para cada historia de usuario que requiera ser priorizada.

Cuando todas las historias de usuario hayan sido priorizadas en el PRD, el *Product Owner* seleccionará las historias determinadas como imprescindibles y las incluirá en el *product backlog*. Este conjunto de historias de usuario representa el mínimo producto viable de la solución. Posteriormente, si el equipo *Scrum* cuenta con la capacidad de completar el mínimo producto viable en la siguiente iteración, se analizará si es posible incluir aquellas historias determinadas como importantes y deseables.

5.2.5.6 Salida: Documento de requisitos del producto (PRD) actualizado

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.5.7 Salida: *Product Backlog* actualizado

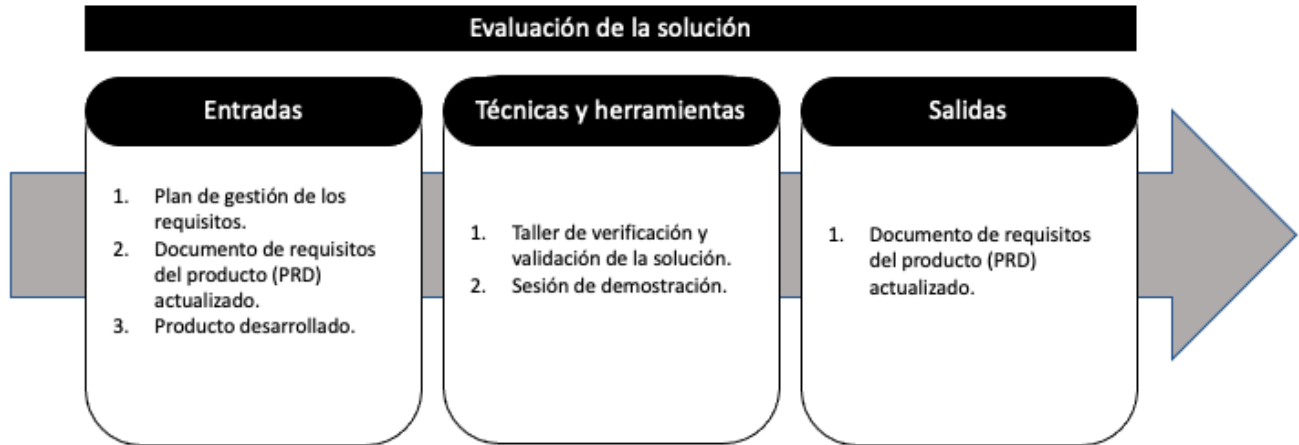
El *product backlog* consiste en el conjunto de historias de usuario bien formuladas, descompuestas, estimadas y priorizadas que se encuentran en el software Jira. Este conjunto de historias de usuario representa el producto de la gestión de requisitos y constituye el trabajo a desarrollarse en las futuras iteraciones.

El *product backlog* es dinámico y podrá sufrir cambios en cada iteración, por lo que debe ser siempre visible para los interesados y miembros del equipo *Scrum*. Cualquier cambio realizado al *product backlog* debe provenir del PRD y ser acorde a él.

5.2.6 *Proceso de evaluación de la solución*

El proceso de evaluación de la solución, mostrado en la Figura 5.9 tiene como objetivo definir si el producto desarrollado cumple con los requisitos y satisface las expectativas de los interesados, con el fin de identificar si es idóneo para su puesta en producción o si se requiere de modificación de los requisitos para futuras iteraciones.

Figura 5.9. Proceso de evaluación de la solución



5.2.6.1 Entrada: Plan de gestión de los requisitos

El plan de gestión de los requisitos es expuesto en la sección 5.2.3.4.

5.2.6.2 Entrada: Documento de requisitos del producto (PRD)

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.6.3 Entrada: Producto desarrollado

El producto desarrollado consiste en el entregable del proceso de desarrollo de software correspondiente a la iteración, el cual se considera funcional y que puede potencialmente ser puesto en producción.

5.2.6.4 Técnica y herramienta: Taller de verificación y validación de la solución

El taller de verificación y validación de la solución tiene como objetivo obtener retroalimentación del *Product Owner* con respecto al producto desarrollado, con el fin de identificar si es idóneo para su puesta en producción o se requiere de modificación de sus requisitos para futuras iteraciones.

El taller consiste en una sesión de trabajo exclusiva del equipo *Scrum*. Durante la sesión, el facilitador seleccionará cada uno de los requisitos desarrollados durante el *sprint* y lo leerá al resto de participantes. Los *developers* demostrarán que la solución cumple el requisito expuesto y sus criterios de aceptación. Posteriormente, se analizará si la solución actual satisface las expectativas de los interesados o si es necesario esperar a futuros *sprints* para lograrlo.

Finalmente, el *Product Owner* brindará su retroalimentación aclarando si el cumplimiento es el esperado o si se requiere modificar el requisito. Si se necesita una modificación al requisito, se incluirá en la sección de registro de cambios del PRD. Dicho cambio será discutido durante el siguiente proceso de obtención de requisitos.

5.2.6.5 Técnica y herramienta: Sesión de demostración

La sesión de demostración tiene como objetivo obtener retroalimentación de los usuarios e interesados clave con respecto al producto desarrollado, con el fin de determinar si realmente satisface sus necesidades.

La sesión será realizada únicamente si el producto desarrollado durante el último *sprint* es funcional y es idóneo para su puesta en producción. En la sesión participará el equipo *Scrum*, los interesados clave y demás invitados que se consideren necesarios para alcanzar el objetivo. Durante la sesión, el facilitador mostrará los requisitos que fueron satisfechos durante la última iteración. Los *developers* mostrarán la solución eficaz haciendo énfasis en las funcionalidades más recientes.

Posteriormente, los interesados brindarán su retroalimentación con respecto al producto desarrollado. Cualquier cambio solicitado, será incluido por el *Product*

Owner en la sección de registro de cambios del PRD. Dicho cambio será discutido durante el siguiente proceso de obtención de requisitos.

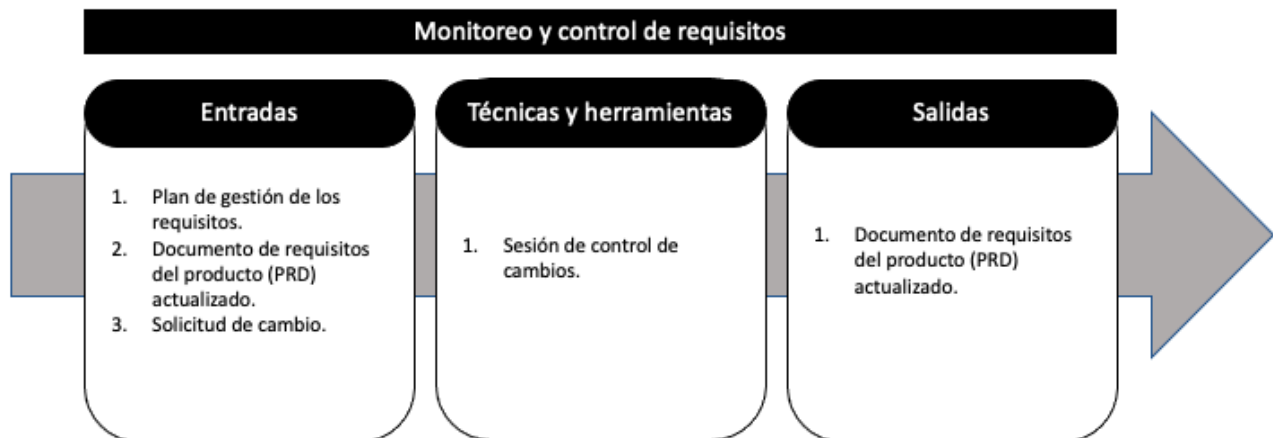
5.2.6.6 Salida: Documento de requisitos del producto (PRD)

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.7 Proceso de monitoreo y control de los requisitos

El proceso de monitoreo y control de los requisitos, mostrado en la Figura 5.10 tiene como objetivo controlar el alcance del *sprint* mediante el monitoreo de los requisitos en desarrollo y el control de cambio de los requisitos.

Figura 5.10. Proceso de monitoreo y control de requisitos



5.2.7.1 Entrada: Plan de gestión de los requisitos

El plan de gestión de los requisitos es expuesto en la sección 5.2.3.4.

5.2.7.2 Entrada: Documento de requisitos del producto (PRD) actualizado

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.2.7.3 Entrada: Solicitud de cambio

La solicitud de cambio consiste en una petición formal de modificación de uno o varios de los requisitos que se encuentran en desarrollo durante el *sprint*, ya sea de forma verbal o escrita. Esta solicitud puede provenir de parte del *Product Owner* o cualquier otro interesado, pero debe ser siempre canalizada mediante el *Product Owner*, quien interpretará si dicha solicitud debe llegar al equipo *Scrum* de inmediato o puede esperar al próximo proceso de obtención de requisitos.

5.2.7.4 Técnica y herramienta: Sesión de control de cambios

La sesión de control de cambios tiene como objetivo controlar el alcance del *sprint* mediante el monitoreo del avance en el desarrollo de los requisitos y el análisis de las solicitudes de cambio. Esta sesión será ejecutada diariamente mientras se mantenga en realización el *sprint*, en ella participará exclusivamente el equipo *Scrum* y consistirá en verificar que los requisitos en desarrollo se mantengan acordes a lo establecido en el PRD y dentro del alcance del *sprint*.

Durante la sesión, el *Product Owner* presentará las solicitudes de cambio que haya recibido. El equipo *Scrum*, evaluará el impacto del cambio solicitado en el alcance del *sprint*. Si el cambio compromete el alcance, el *Product Owner* lo documentará en la sección de registro de cambios del PRD y lo presentará durante el siguiente proceso de obtención de requisitos programado en el plan de gestión de requisitos. Si el cambio solicitado no compromete el alcance o si la urgencia del cambio es alta, inmediatamente después de la finalización de la sesión, se realizará un proceso de obtención y análisis de requisitos extraordinario, exclusivamente para las necesidades que deben modificarse. Durante la sesión, el equipo *Scrum* podrá incluir

consultas en la sección de preguntas generadoras del PRD, para ser analizadas durante el siguiente proceso de obtención de requisitos programado.

5.2.7.5 Salida: Documento de requisitos del producto (PRD) actualizado

El documento de requisitos del producto es expuesto en la sección 5.2.3.5.

5.3 Diccionario de términos para la gestión de requisitos

El diccionario de términos tiene como objetivo unificar todos aquellos términos de la gestión de requisitos y el marco de trabajo *Scrum* para fomentar la adopción de un lenguaje común entre los practicantes de la guía metodológica. El diccionario será adjuntado a la guía y servirá de insumo al lector para aclarar conceptos y utilizar la terminología adecuada. En el apéndice O se muestra el diccionario de términos para la gestión de requisitos.

5.4 Estrategia de implementación del método de gestión de requisitos

La estrategia de implementación tiene como objetivo brindar a la Tribu Navi un estimado del alcance, tiempo y costo de la puesta en marcha de la propuesta de solución a su problemática. Esta estrategia parte del hecho de que los *Scrum Master* de la Tribu Navi son los responsables de la implementación y seguimiento de la solución.

5.4.1 Fases y actividades de la implementación

El plan de implementación de la propuesta consta de cinco fases, las cuales se detallan a continuación.

5.4.1.1 Primera fase: Sensibilización del personal

El objetivo de esta fase consiste en crear conciencia en los líderes y demás miembros de la Tribu Navi sobre la problemática identificada y la importancia de la

solución propuesta, así como los beneficios que se desean alcanzar. Durante esta fase se realizarán las siguientes actividades:

- Sesión inicial: Los *Scrum Master* presentarán la propuesta de solución al *Chief Product Owner* y al *Chief Scrum Master*. Durante esta sesión se hará énfasis en la descripción de la problemática, los resultados obtenidos para su identificación, la importancia de la gestión de requisitos, los componentes de la propuesta de solución y los beneficios esperados. Además, se hará énfasis en la importancia de acatar el método propuesto de forma obligatoria para todos los equipos *Scrum*. Deben ser aclaradas todas las dudas por parte de los participantes y obtener el visto bueno para la aplicación del resto de fases según el cronograma propuesto más adelante.

- Comunicación de la estrategia de implementación: Los *Scrum Master* formularán comunicados de sensibilización para el resto de los miembros de la tribu. Estos deben contener material gráfico que exponga la importancia del método de gestión de requisitos de proyectos y su estrategia de implementación. Asimismo, deben asegurar que los comunicados son recibidos y comprendidos por los miembros de los equipos *Scrum*. Esta actividad se extiende por dos semanas.

5.4.1.2 Segunda fase: Fortalecimiento de las competencias

El objetivo de esta fase es identificar el estado actual de las competencias individuales de los miembros de los equipos *Scrum* y fortalecer aquellas que no sean idóneas para desempeñar su rol en la gestión de requisitos. En esta fase se realizarán las siguientes actividades:

- Análisis de competencias: Los *Scrum Master* definirán un conjunto de competencias relacionadas a la gestión de requisitos de proyectos y una escala

cuantitativa que les permita determinar el estado actual y el estado deseado de cada una de ellas. Posteriormente, medirán el estado actual de dichas competencias para cada miembro del equipo *Scrum* e identificarán las brechas que los separan de su estado idóneo. Esta actividad tiene una duración de dos semanas.

- Capacitación de la gestión de requisitos: Los *Scrum Master* capacitarán a todos los miembros de la Tribu Navi sobre el nuevo método de gestión de requisitos, incluyendo el flujograma de procesos, la guía metodológica y, el diccionario de términos. Esta actividad tiene una duración de ocho horas divididas en dos sesiones de cuatro horas cada una.

- Talleres de fortalecimiento de las competencias: Los *Scrum Master* realizarán talleres enfocados en el fortalecimiento de las competencias específicas para la gestión de requisitos. Estos talleres serán dirigidos por los miembros con mayor desarrollo de sus competencias y serán invitados todos aquellos miembros que requieran fortalecerlas. Esta actividad se conforma de sesiones de dos horas para cada competencia que requiera ser fortalecida y se extenderá por dos semanas.

5.4.1.3 Tercera fase: Implementación asistida

El objetivo de esta fase es implementar el método de gestión de requisitos en un único proyecto por equipo *Scrum*. Durante esta fase los *Scrum Master* asistirán en todo momento a su equipo, verificarán la correcta aplicación del método, la adaptabilidad del personal al cambio y darán asesoría. En esta fase se realizarán las siguientes actividades:

- Kickoff: Los *Scrum Master* realizarán una sesión de inicio de implementación con sus equipos correspondientes, en la cual se definirá un único proyecto que será parte del nuevo método de gestión.

- Seguimiento: Los *Scrum Master* vigilarán el cumplimiento del nuevo método durante cada *sprint*, siendo los facilitadores de cada una de las sesiones y talleres establecidos en la guía metodológica.

- Retrospectiva: Los *Scrum Master* realizarán una sesión de retrospectiva con sus equipos al final de cada *sprint*, en la cual se harán observaciones sobre lo que el equipo está haciendo correctamente y lo que no.

Se prevé que esta fase se extienda por 12 semanas, tiempo suficiente para obtener una retroalimentación consistente del equipo *Scrum*. Aun así, quedará a criterio del *Scrum Master* extender la implementación asistida por más tiempo.

5.4.1.4 Cuarta fase: Evaluación

El objetivo de esta fase es evaluar si el método ha sido correctamente adoptado y si muestra resultados positivos con relación a la mitigación de la problemática. Esta fase tiene una duración de 2 semanas y en ella se realizarán las siguientes actividades:

- Análisis de retrospectiva: Los *Scrum Master* realizarán un análisis de la información obtenida durante las retrospectivas de la fase de implementación asistida.

- Análisis de beneficios: Los *Scrum Master* medirán el impacto que ha tenido el método en la disminución del trabajo no completado por *sprint* y sus efectos.

- Ajustes en el método: Los *Scrum Master* discutirán, ejecutarán y comunicarán los ajustes al método que consideren adecuados.

Si los *Scrum Master* determinan que un equipo *Scrum* ha adoptado el método de forma adecuada, procederá a la implementación total con dicho equipo. De lo contrario, se mantendrá en ejecución asistida.

5.4.1.5 Quinta fase: Implementación total

El objetivo de esta fase final es implementar el modelo en la totalidad de proyectos del equipo *Scrum* sin requerir de asistencia por parte del *Scrum Master*.

Durante la implementación total, el equipo *Scrum* será responsable de gestionar los requisitos de proyectos sin la necesidad de ser asistidos por el *Scrum Master* y utilizará sus eventos *Scrum* para hacer inspección y adaptación de sus procesos internos. Si durante esta fase final, el equipo identifica una oportunidad de mejora, ésta debe ser presentada al resto de *Scrum Master* para ser evaluada y ajustar el método manteniendo la estandarización.

5.4.2 Presupuesto

En la Tabla 5.1., se observan los costos correspondientes a la implementación del método propuesto por fase y los montos totales. Para el cálculo de costos de las labores realizadas por los *Scrum Master* como implementadores, se tomó como referencia el costo unitario de la hora profesional de un especialista en metodologías ágiles.

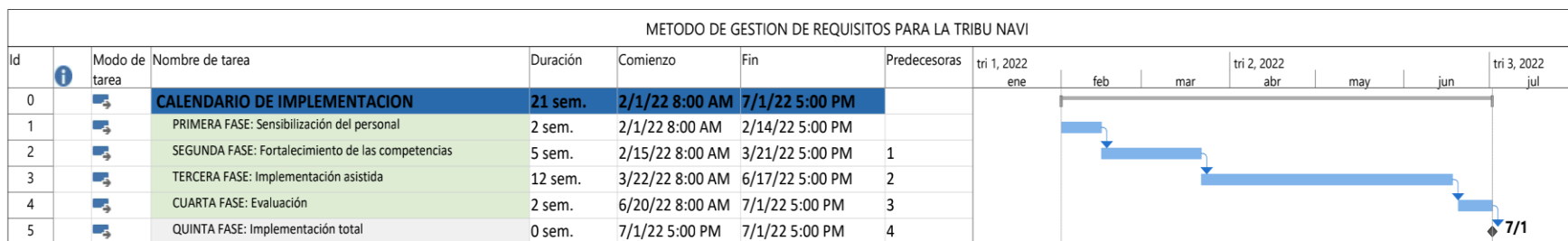
Tabla 5.1. Presupuesto de la implementación

Fase	Hora profesional	Costo unitario	Subtotal
Sensibilización del personal	8	\$80.00	\$640.00
Fortalecimiento de las competencias	24	\$80.00	\$1920.00
Implementación asistida	24	\$80.00	\$1920.00
Evaluación	16	\$80.00	\$1280.00
Implementación total	0	\$80.00	\$0.00
Total	72		\$5760.00

5.4.3 Cronograma

En la **¡Error! La autoreferencia al marcador no es válida.**, se detalla la duración de cada fase de la implementación. La fecha de inicio está propuesta para el día 01 de febrero de 2022 y su finalización para el 04 de julio de 2022, para un total de 21 semanas. Se considera que la organización labora de lunes a viernes con un horario de 8:00 a.m. a 5:00 p.m. y aplica los días feriados de ley.

Figura 5.11. Cronograma de implementación



Capítulo 6. Conclusiones y Recomendaciones

En este capítulo se exponen las conclusiones y recomendaciones del estudio.

6.1 Conclusiones

A continuación, se muestran las conclusiones del estudio:

- Los resultados obtenidos permiten concluir que las prácticas actuales de gestión de requisitos utilizadas por los equipos *Scrum*:
 - o No cuentan con un proceso formal y no generan entregables definidos.
 - o Se realizan durante los eventos de *Scrum* de forma aislada y sin una secuencia lógica.
 - o No son siempre utilizadas ni están estandarizadas entre equipos.
 - o Establecen lo que debe hacerse, pero no establecen el cómo hacerlo.

- Las prácticas de gestión de requisitos actuales han incidido directamente en el incumplimiento de las entregas, provocando retrasos en los proyectos, incumplimiento de los compromisos y disminuyendo el rendimiento de los equipos *Scrum*.

- El Manual Operativo *Scrum* actual es un recurso que extrae de la Guía *Scrum* las prácticas de gestión de requisitos más comunes (identificación, documentación, estimación, descomposición, priorización, revisión y control de requisitos), pero no define los procesos, técnicas y herramientas a utilizar para una correcta gestión.

- Las competencias relacionadas a la gestión de requisitos en la Tribu Navi están desniveladas entre equipos, por lo que la sensibilización del personal, la capacitación y la transferencia de conocimiento, generarán el fortalecimiento de las competencias necesario para gestionar adecuadamente los requisitos según el método propuesto.

- Los resultados obtenidos permiten concluir que las buenas prácticas identificadas para la evaluación de la necesidad, evaluación de la solución, planificación de la gestión,

obtención, análisis y control de requisitos son idóneas para la mitigación de la problemática de la Tribu Navi, ya que:

- Proveen soluciones a cada elemento de la problemática de la Tribu Navi.
 - Formalizan y fortalecen las prácticas actuales.
 - Se adaptan al marco de trabajo *Scrum* y los principios ágiles.
 - Se adaptan a Jira como herramienta de gestión.
 - Son reconocidas y avaladas mundialmente.
- Es posible concluir que el método de gestión de requisitos propuesto, conformado por el flujograma de procesos, la guía metodológica y el diccionario de términos para la gestión de requisitos, es idóneo para solventar la problemática de la Tribu Navi, ya que:
- Solventa las principales causas de la problemática de la Tribu Navi (complejidad, ambigüedad, dependencia de terceros, dependencia entre requisitos, cambios de prioridad y cambios en los requisitos).
 - Es acorde al marco de trabajo *Scrum* y los principios ágiles.
 - Favorece la inspección y adaptación mediante procesos iterativos.
 - Favorece la estandarización de los procesos en la Tribu Navi.
 - Es un recurso de la organización que puede utilizarse en otras tribus, siempre y cuando se realicen las adaptaciones necesarias y se gestione adecuadamente el cambio.
- La estrategia de implementación conformada por cinco fases (sensibilización, fortalecimiento de las competencias, implementación asistida, evaluación e implementación total) facilita la adopción del método de gestión en la Tribu Navi y favorece la adaptación al cambio. Esta implementación tiene una duración de 21 semanas y representa un costo estimado de \$5760.

6.2 Recomendaciones

A continuación, se muestran las recomendaciones del estudio:

- Se recomienda a los *Scrum Master* incorporar el indicador de trabajo comprometido vs. trabajo no completado en sus métricas de seguimiento, de forma que sea analizado en cada sesión de *sprint retrospective* y permita identificar si el equipo *Scrum* mejora o desmejora su rendimiento.
- Se recomienda a los *Scrum Master* planificar y ejecutar capacitaciones periódicas anuales de actualización para sus equipos *Scrum*, que incluyan los principios del manifiesto ágil, el marco de trabajo *Scrum*, el Manual Operativo *Scrum* de la organización y el nuevo método de gestión de requisitos.
- Se recomienda a los *Scrum Master* realizar un análisis de competencias individuales relacionadas a la gestión de requisitos cuando ingresen nuevos miembros a los equipos *Scrum*. Una vez identificadas, se recomienda realizar sesiones de transferencia de conocimiento, dirigidas por los miembros cuyas competencias estén más desarrolladas.
- Se recomienda al *Chief Scrum Master* realizar un estudio de cargas de trabajo de los *Scrum Master* que le permita determinar si cuentan con la disponibilidad suficiente para llevar a cabo sus labores rutinarias y a la vez cumplir con el cronograma de implementación.
- Se recomienda a los *Product Owner* formalizar las actividades de cierre del proyecto, incluyendo la documentación de lecciones aprendidas para la transferencia de conocimiento, el soporte de la transición del producto a operaciones y la habilitación del mantenimiento de beneficios a largo plazo.
- Para proyectos fuertemente regulados, de gran complejidad o alto riesgo, se recomienda incorporar una pizarra de control de cambios. Esta pizarra consiste en un grupo de interesados responsables por revisar, evaluar, aprobar, retrasar, rechazar y comunicar cambios en los requisitos y en el proyecto.

Capítulo 7. Referencias bibliográficas

Atlassian (01 de noviembre de 2018). *Using JIRA for Requirements Management*.
<https://confluence.atlassian.com/jirakb/using-jira-for-requirements-management-193300521.html>

Atlassian (01 de noviembre de 2018). *How to do Scrum with JIRA software*.
<https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software>

Atlassian (01 de noviembre de 2018). *Hojas de ruta de productos*.
<https://www.atlassian.com/es/agile/product-management/product-roadmaps>

Atlassian (01 de noviembre de 2018). *Plantilla para los requisitos de productos*.
<https://www.atlassian.com/es/software/confluence/templates/product-requirements>

Cohn, M. (2004). *User Stories Applied for Agile Software Development*. Pearson Education. Boston, Massachusetts.

Hernández, R. (2014). *Metodología de la investigación, Sexta edición*. McGraw-Hill. México.

IAG Consulting (2009). *Business Analysis of Benchmark: The Path to Success*. New Castle, Delaware.

Infosys (2019). *Model-Based Systems Engineering: Digitized Requirements Management for Aerospace and Defense Products*. Infosys Limited. Bengaluru, India.

Institute of Electrical and Electronics Engineers (1998). *Guide for Developing System Requirements Specifications*. New York, New York.

Institute of Electrical and Electronics Engineers (1998). *Recommended Practice for Software Requirements Specifications*. New York, New York.

NASA (2007). *Systems Engineering Handbook*. NASA Headquarters, Room 6D37. Washington, DC.

Niño, V. (2011). *Metodología de la Investigación Diseño y Ejecución*. Ediciones de la U. Bogotá, Colombia.

Project Management Institute (2017). *A guide to the project management body of knowledge PMBOK® Guide Sixth edition*. Newtown Square, Pensilvania.

Project Management Institute (2017). *Agile Practice Guide*. Newtown Square, Pensilvania.

Project Management Institute (2017). *Lexicon of Project Management Term*. Newtown Square, Pensilvania.

Project Management Institute (2014). *Pulso de la profesión®: Gestión de Requisitos: Una competencia esencial para el éxito de proyectos y programas*. Newtown Square, Pensilvania.

Project Management Institute (2016). *Requirement Management: A Practice Guide*. Newtown Square, Pensilvania.

Schwaber, K. y Sutherland, J. (2020). *La Guía de Scrum*.

U.S. Department of Transportation (2009), *Requirements Engineering Management Handbook*. Rockwell Collins, Inc. Washington, DC.

Young, R. (2004). *The Requirements Engineering Handbook*. Artech House. Boston, Massachusetts.

Capítulo 8 Apéndices

8.1 Apéndice A. Cuestionario de diagnóstico

Cuestionario de diagnóstico: Gestión de Requisitos

Este cuestionario tiene como objetivo identificar la frecuencia con la que son utilizadas las practicas relacionadas a la gestión de requisitos de proyectos en los equipos Scrum. Sus resultados serán utilizados exclusivamente con fines académicos.

Cada pregunta cuenta con una escala del 1 al 5, la cual debe interpretarse como se muestra a continuación:

- 1- NUNCA
- 2- CASI NUNCA
- 3- OCASIONALMENTE
- 4- CASI SIEMPRE
- 5- SIEMPRE

GLOSARIO:

Requisito: Funcionalidad requerida por el usuario que se desarrolla durante el sprint.

Historia de Usuario: Descripción en prosa del requisito expresado desde la perspectiva del usuario.

INVEST: Criterio que define que un requisito debe ser individual, negociable, valioso, estimable, pequeño y testeable.

Jira: software de gestión de proyectos que permite documentar los requisitos.

Sección 1. Gestión de requisitos

Durante el inicio del proyecto, ¿se identifica la necesidad planteada? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se identifica a los interesados que generan los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se obtienen los requisitos directamente de los interesados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se realizan prototipos de la solución? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se documentan los requisitos en el software JIRA? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se descomponen los requisitos en otros más discretos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se estima el esfuerzo de desarrollar cada requisito? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se clasifican y organizan los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se priorizan los requisitos según el valor de negocio que generan? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se genera una línea base de requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el inicio del proyecto, ¿se obtiene la aprobación de la línea base por parte de los interesados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el proceso de desarrollo, ¿los cambios en los requisitos son documentados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el proceso de desarrollo, ¿los cambios en los requisitos se comunican a los interesados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el proceso de desarrollo, ¿los cambios en los requisitos son aprobados por los interesados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el proceso de desarrollo, ¿se obtiene retroalimentación por parte del Product Owner en cada sprint? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el proceso de desarrollo, ¿se obtiene retroalimentación por parte de los interesados en cada sprint? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el cierre del proyecto, ¿se valida que la solución satisface la necesidad planteada? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante el cierre del proyecto, ¿se obtiene la aprobación formal de la solución por parte de los interesados? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Sección 2. Formulación de requisitos

Durante la formulación de requisitos, ¿se incluye una descripción clara del requisito que evita la ambigüedad? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se utiliza la técnica de historia de usuario para la redacción de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se utiliza el método INVEST para la redacción de las historias de usuario? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se describen las entradas y salidas del sistema de software? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se describen las funciones que ejecutará el software mediante el procesamiento de entradas y la generación de salidas? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se especifican los requisitos de desempeño del software? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se especifican los requisitos de base de datos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Durante la formulación de requisitos, ¿se especifican las restricciones de diseño? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

Sección 3. Competencias de gestión de requisitos

¿Cuenta el equipo con conocimiento teórico para la identificación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con experiencia práctica para la identificación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con conocimiento teórico para la documentación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con experiencia práctica para la documentación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con conocimiento teórico para la estimación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con experiencia práctica para la estimación de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con conocimiento teórico para la priorización de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con experiencia práctica para la priorización de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con conocimiento teórico para el control de cambio de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

¿Cuenta el equipo con experiencia práctica para el control de cambio de los requisitos? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Siempre

8.2 Apéndice B. Guía de entrevista

Guía de Entrevista: Gestión de Requisitos

Esta entrevista tiene como objetivo profundizar en las prácticas relacionadas a la gestión de requisitos de proyectos en los equipos Scrum.

Está dirigida a los Scrum Master.

Sus resultados serán utilizados exclusivamente con fines académicos.

A continuación, se presentan preguntas abiertas relacionadas a la gestión de requisitos de proyectos. Cada pregunta debe ser contestada ampliamente, haciendo énfasis en los detalles relacionados a los procesos, técnicas y herramientas utilizadas. Es posible adjuntar material gráfico que ejemplifique las respuestas.

* Required

¿Cómo se identifican los requisitos? *

Your answer

¿Cómo se documentan los requisitos? *

Your answer

¿Cómo se estiman los requisitos? *

Your answer

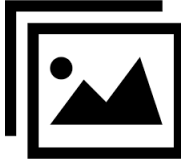
¿Cómo se priorizan los requisitos? *

Your answer

¿Cómo se controla el cambio en los requisitos? *

Your answer

8.3 Apéndice C. Formato de ficha de síntesis

Identificador:	P-000
Nombre de la práctica:	Práctica de Ejemplo
Descripción de la Práctica:	Descripción en prosa  Imágenes
Elementos de la problemática a los que se relaciona o solventa:	Lista de elementos - -
Fuente de información:	Fuente de la cuál fue tomada la práctica

8.4 Apéndice D. Formato de inventario

Título del Inventario			
Identificador de la práctica	Nombre de la práctica	Categoría	Elementos de la problemática a los que se relaciona o solventa
P-001			
P-002			
P-003			
...

8.5 Apéndice E. Formato de lista de verificación

Lista de verificación de documentación de requisitos

Equipo Scrum

Your answer

Identificador del requisito

Your answer

Historia de Usuario

- Rol
- Expectativa
- Razón

Criterio INVEST

- Independiente
- Negociable
- Valorable
- Estimable
- Pequeño
- Testeable

Información complementaria

- Representación gráfica (diagrama o esquema)
- Material visual de apoyo (imagen o video)
- Material documental (guía, manual o lineamiento)
- Other: _____

Comentarios

Your answer _____

8.6 Apéndice F. Fichas de síntesis del Manual Operativo Scrum

Identificador:	F-001
Nombre de la práctica:	Identificación de la necesidad
Descripción de la Práctica:	El <i>Product Owner</i> es el responsable de desarrollar y comunicar explícitamente el objetivo del producto, así como de crear y comunicar los elementos del <i>Product Backlog</i> .
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas.
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

Identificador:	F-002
Nombre de la práctica:	Control de cambios de requisitos
Descripción de la Práctica:	El <i>Product Owner</i> es el responsable de gestionar las solicitudes de incorporación, modificación y eliminación de elementos del <i>Product Backlog</i> .
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Control de cambios - Prioridad durante el <i>sprint</i>
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

Identificador:	F-003														
Nombre de la práctica:	Documentación de requisitos														
Descripción de la Práctica:	<p>El <i>Product Owner</i> es el responsable de documentar los criterios de calidad y aceptación para cada elemento del <i>Product Backlog</i>.</p> <p>El equipo <i>Scrum</i> debe documentar los criterios que permiten dejar listos los elementos del <i>Product Backlog</i> para su ejecución y los criterios necesarios para aceptarlos como concluidos.</p> <p>Se debe documentar la evidencia del cumplimiento de estos criterios en el <i>software</i> JIRA. Toda historia de usuario debe contar con los siguientes criterios "INVEST" para ser incluida en un <i>sprint</i>.</p> <table border="1" data-bbox="609 594 1295 1035"> <thead> <tr> <th>Criterio</th> <th>Descripción</th> </tr> </thead> <tbody> <tr> <td>Inmediatamente accionable:</td> <td>¿Está libre de bloqueos externos al equipo? ¿Se puede entregar de forma independiente?</td> </tr> <tr> <td>Negociable:</td> <td>¿Suficientemente descriptiva para apoyar el debate y la conversación del equipo?</td> </tr> <tr> <td>Valiosa:</td> <td>¿Entrega valor visible a los clientes?</td> </tr> <tr> <td>Estimable:</td> <td>¿Suficientemente clara para que el equipo la pueda estimar?</td> </tr> <tr> <td>"sized to fit" Tamaño correcto para el sprint</td> <td>Dividida en bloques pequeños para poder terminarla dentro del <i>sprint</i>.</td> </tr> <tr> <td>"Testable" (Testable)</td> <td>Criterios de aceptación claros para saber que es "un resultado aceptable"</td> </tr> </tbody> </table>	Criterio	Descripción	Inmediatamente accionable:	¿Está libre de bloqueos externos al equipo? ¿Se puede entregar de forma independiente?	Negociable:	¿Suficientemente descriptiva para apoyar el debate y la conversación del equipo?	Valiosa:	¿Entrega valor visible a los clientes?	Estimable:	¿Suficientemente clara para que el equipo la pueda estimar?	"sized to fit" Tamaño correcto para el sprint	Dividida en bloques pequeños para poder terminarla dentro del <i>sprint</i> .	"Testable" (Testable)	Criterios de aceptación claros para saber que es "un resultado aceptable"
Criterio	Descripción														
Inmediatamente accionable:	¿Está libre de bloqueos externos al equipo? ¿Se puede entregar de forma independiente?														
Negociable:	¿Suficientemente descriptiva para apoyar el debate y la conversación del equipo?														
Valiosa:	¿Entrega valor visible a los clientes?														
Estimable:	¿Suficientemente clara para que el equipo la pueda estimar?														
"sized to fit" Tamaño correcto para el sprint	Dividida en bloques pequeños para poder terminarla dentro del <i>sprint</i> .														
"Testable" (Testable)	Criterios de aceptación claros para saber que es "un resultado aceptable"														
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos 														
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.														

Identificador:	F-004
Nombre de la práctica:	Descomposición de requisitos
Descripción de la Práctica:	<p>Los <i>Developers</i> descomponen los elementos del <i>Product</i> en unidades de trabajo de un día o menos.</p> <p>Las historias de alto nivel deben ser descompuesta de manera que sea posible entregar valor en cada <i>sprint</i>.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos - Dependencia entre requisitos
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

Identificador:	F-005
Nombre de la práctica:	Revisión de requisitos
Descripción de la Práctica:	<p>Durante la finalización del <i>sprint</i>, en una sesión de aproximadamente 2 horas, el equipo <i>Scrum</i> se reúne para inspeccionar los requisitos trabajados. Esto facilita la retroalimentación y la colaboración.</p> <p>La sesión debe asistir el equipo <i>Scrum</i> y aquellos interesados clave que hayan sido invitados por el <i>Product Owner</i>.</p> <p>El equipo <i>Scrum</i> muestra a los interesados los requisitos completados y aclara preguntas. Se analiza en conjunto si la necesidad ha cambiado, si surgieron nuevas oportunidades y se ajusta el plan de acción.</p> <p>Al finalizar la sesión, se obtiene una lista de requisitos actualizada y revisada.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Cambios no gestionados en los requisitos.
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

Identificador:	F-006																		
Nombre de la práctica:	Estimación de requisitos																		
Descripción de la Práctica:	<p>Todos los requisitos del Product Backlog deben contar con un valor de esfuerzo estimado en puntos, nunca en tiempo</p> <p>La estimación debe ser realizada por quienes realizarán el trabajo.</p> <p>La estimación debe realizarse durante las sesiones de refinamiento.</p> <p>La estimación debe estar basada en los valores de talla del siguiente cuadro.</p> <table border="1" data-bbox="604 1213 1409 1503"> <thead> <tr> <th>Talla</th> <th>Puntos Historia</th> <th>Puntos Epic</th> </tr> </thead> <tbody> <tr> <td>XS</td> <td>2</td> <td>13</td> </tr> <tr> <td>S</td> <td>3</td> <td>21</td> </tr> <tr> <td>M</td> <td>5</td> <td>35</td> </tr> <tr> <td>L</td> <td>8</td> <td>54</td> </tr> <tr> <td>XL</td> <td>13</td> <td>100</td> </tr> </tbody> </table>	Talla	Puntos Historia	Puntos Epic	XS	2	13	S	3	21	M	5	35	L	8	54	XL	13	100
Talla	Puntos Historia	Puntos Epic																	
XS	2	13																	
S	3	21																	
M	5	35																	
L	8	54																	
XL	13	100																	
Elementos de la problemática a los que se relaciona o solventa:	- Complejidad de los requisitos																		
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.																		

Identificador:	F-007
Nombre de la práctica:	Identificación de requisitos
Descripción de la Práctica:	Cualquier interesado puede agregar requisitos al <i>Product Backlog</i> . Deben definirse las principales funcionalidades del producto, las historias de alto nivel, las sub-historias y los usuarios que utilizarán el producto.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

Identificador:	F-008
Nombre de la práctica:	Priorización de requisitos
Descripción de la Práctica:	El <i>Product Owner</i> debe priorizar constantemente adaptándose al aprendizaje generado por la retroalimentación de los interesados. Se deben priorizar los requisitos que generan mayor valor en el menor tiempo.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	Manual Operativo <i>Scrum</i> de la Tribu Navi.

8.7 Apéndice G. Resultados de lista de verificación

Equipo Scrum	Identificador del requisito	Elemento de Historia de Usuario	Criterio INVEST	Información complementaria	Comentarios
OMA	OMA-2958	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		Rol: "Como usuario de negocio requiero"
OMA	OMA-3285	Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3304	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3352	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	Rol: ":Como usuario de desarrollo"
OMA	OMA-3354				
OMA	OMA-3355				
OMA	OMA-2870	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	
OMA	OMA-2916	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Representación gráfica (diagrama o esquema), Material documental (guía, manual o lineamiento)	Rol: "Como usuario de desarrollo"
OMA	OMA-3356	Expectativa, Razón	Independiente, Negociable, Valorable		Expectativa: "Requiero realizar una prueba integral"
OMA	OMA-3357				
OMA	OMA-3360				
OMA	OMA-3367	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3372	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3388	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3393	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3400	Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3402	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	Rol: "Como usuario de desarrollo"
OMA	OMA-2870	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	
OMA	OMA-2916	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Representación gráfica (diagrama o esquema), Material documental (guía, manual o lineamiento)	Rol: "Como usuario de desarrollo"
OMA	OMA-3358	Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3383	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-2870	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-2916	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Representación gráfica (diagrama o esquema), Material documental (guía, manual o lineamiento)	
OMA	OMA-3358	Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3378	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		

OMA	OMA-3383	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3404	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3407	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3408				
OMA	OMA-3411	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Usuario de Negocio"
OMA	OMA-3435	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3446	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3451				
OMA	OMA-3471	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	
OMA	OMA-3425	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3440	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3459	Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-2873	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2905	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2906	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2907	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2908	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2909	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2910	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2911	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3418	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3403	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3409	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3409	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3425	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3430	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3440	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3459	Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3461				Rol: "Como usuario de desarrollo"
OMA	OMA-3474	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		

OMA	OMA-3475	Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3480	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3484	Expectativa			Expectativa: "FIX de S&F en MPI y pruebas exitosas"
OMA	OMA-2959				
OMA	OMA-2960				
OMA	OMA-2961				
OMA	OMA-3460				
OMA	OMA-2959				
OMA	OMA-2960				
OMA	OMA-3460				
OMA	OMA-3462	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3463	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3485				
OMA	OMA-3487	Expectativa			
OMA	OMA-3488	Expectativa			
OMA	OMA-3489				
OMA	OMA-3490				
OMA	OMA-3493				
OMA	OMA-3495				
OMA	OMA-3502	Expectativa			
OMA	OMA-3505	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3515	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3526	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3531				
OMA	OMA-3534	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-2961				
OMA	OMA-3359				
OMA	OMA-3486			Material documental (guía, manual o lineamiento)	
OMA	OMA-3510	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3520	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3532	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3533	Expectativa			

OMA	OMA-3540	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3541				
OMA	OMA-3542	Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3544				
OMA	OMA-3545				
OMA	OMA-3546				
OMA	OMA-3549	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	
OMA	OMA-3550				
OMA	OMA-3465	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3466	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3467	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3468	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3473	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3551	Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3563				
OMA	OMA-3582	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3583				
OMA	OMA-3614	Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3591				
OMA	OMA-3615	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3491	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3570	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3575	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3591				
OMA	OMA-3598	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3603	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3609	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3618	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3619	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3620	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3627				

OMA	OMA-3628	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		Rol: "Como usuario de negocio"
OMA	OMA-3631				
OMA	OMA-3632	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		Rol: "Como Usuario de negocio"
OMA	OMA-3633				
OMA	OMA-3615	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño	Material documental (guía, manual o lineamiento)	
OMA	OMA-3472	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3504				
OMA	OMA-3547	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Pequeño		
OMA	OMA-3548	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Pequeño		
OMA	OMA-3626	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		
OMA	OMA-3665	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3733				
OMA	OMA-3735	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3736	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como usuario de negocio"
OMA	OMA-3737	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como usuario de negocio"
OMA	OMA-3738	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3739	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3740	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		Rol: "Como usuario desarrollador"
OMA	OMA-3743	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño	Representación gráfica (diagrama o esquema)	Rol: "Como usuario desarrollador"
OMA	OMA-3744				
OMA	OMA-3732	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		Rol: "Como usuario de negocio"
OMA	OMA-2962				
OMA	OMA-3727	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
OMA	OMA-3746				
OMA	OMA-3749	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3750	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3751	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
OMA	OMA-3761				
OMA	OMA-3768	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3773				

OMA	OMA-3774				
OMA	OMA-3775	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3780	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3785	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3790	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3795	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3800				
OMA	OMA-3809				
OMA	OMA-3469	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3747	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3748	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3469	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3747	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3748	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3752	Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3806	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3807	Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3811				
OMA	OMA-3812	Rol, Expectativa	Independiente, Negociable, Valorable		
OMA	OMA-3814	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3815	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
OMA	OMA-3816				
OMA	OMA-3753	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
OMA	OMA-3758				
OMA	OMA-3805	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
TAW	TAW-2544	Expectativa			
TAW	TAW-2544				
TAW	TAW-2553				
TAW	TAW-2562				Expectativa: "Avanzar los riesgos tecnológicos identificados"
TAW	TAW-2580			Material visual de apoyo (imagen o video)	
TAW	TAW-2566	Expectativa		Material visual de apoyo (imagen o video)	Expectativa: "Realizar pruebas controladas"

TAW	TAW-2583				
TAW	TAW-2585				
TAW	TAW-2593				
TAW	TAW-2594				
TAW	TAW-2587			Material documental (guía, manual o lineamiento)	
TAW	TAW-2588			Material visual de apoyo (imagen o video)	
TAW	TAW-2586				
TAW	TAW-2588				Expectativa: "Prueba Controlada "
TAW	TAW-2590			Material visual de apoyo (imagen o video)	
TAW	TAW-2591				
TAW	TAW-2592				
TAW	TAW-2611				
TAW	TAW-2470				
TAW	TAW-2587				
TAW	TAW-2595				
TAW	TAW-2470				
TAW	TAW-2595				
TAW	TAW-2612				Expectativa: "Prueba Controlada "
TAW	TAW-2613				Expectativa: "Prueba Controlada"
TAW	TAW-2614				Expectativa: "Prueba controlada"
TAW	TAW-2615				Expectativa: "Prueba Controlada"
TAW	TAW-2617				
TAW	TAW-2584				
TAW	TAW-2616				Expectativa: "Prueba controlada"
TAW	TAW-2620				Expectativa: "Completar la documentacion"
TAW	TAW-2623				
TAW	TAW-2624				
TAW	TAW-2587				
TAW	TAW-2634				
TAW	TAW-2636				
TAW	TAW-2637				

TAW	TAW-2638				
TAW	TAW-2643				
TAW	TAW-2641				
TAW	TAW-2642				
TAW	TAW-2616				Expectativa: "Prueba controlada"
TAW	TAW-2620				
TAW	TAW-2623				
TAW	TAW-2624				
TAW	TAW-2634				
TAW	TAW-2636				
TAW	TAW-2637				
TAW	TAW-2641				
TAW	TAW-2621			Material documental (guía, manual o lineamiento)	
TAW	TAW-2635			Material documental (guía, manual o lineamiento)	
TAW	TAW-2642				
TAW	TAW-2649				Expectativa: "Completar documento"
TAW	TAW-2651				
TAW	TAW-2652				
TAW	TAW-2656				
TAW	TAW-2658				
TAW	TAW-2659				
TAW	TAW-2650				
TAW	TAW-2660			Material documental (guía, manual o lineamiento)	
TAW	TAW-2663				
TAW	TAW-2664				
TAW	TAW-2657				
TAW	TAW-2452	Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
TAW	TAW-2626			Material visual de apoyo (imagen o video)	
TAW	TAW-2640				
TAW	TAW-2675				
TAW	TAW-2676				
TAW	TAW-2578				

TAW	TAW-2625	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
TAW	TAW-2625	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
TAW	TAW-2685				
TAW	TAW-2686	Rol, Expectativa	Independiente, Negociable, Valorable		
TAW	TAW-2687				
TAW	TAW-2578				
TAW	TAW-2576				
TAW	TAW-2577				
TAW	TAW-2657				
TAW	TAW-2690			Material visual de apoyo (imagen o video)	
TAW	TAW-2699			Material visual de apoyo (imagen o video)	
TAW	TAW-2703				Expectativa: "Ejecución de pruebas"
TAW	TAW-2710				
TAW	TAW-2711				
TAW	TAW-2737				
TAW	TAW-2739				
TAW	TAW-2744				
TAW	TAW-2574				
TAW	TAW-2738				
TAW	TAW-2574				
TAW	TAW-2738				
TAW	TAW-2745	Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
TAW	TAW-2746	Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
TAW	TAW-2747				
TAW	TAW-2749				
TAW	TAW-2754				
TAW	TAW-2755				
TAW	TAW-2753				
TAW	TAW-2757				
TAW	TAW-2758				
TAW	TAW-2759				
TAW	TAW-2761				

TAW	TAW-2762				
TAW	TAW-2763				
TAW	TAW-2771				
KEK	KEK-1956	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2261	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Product Owner"
KEK	KEK-2262	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Product Owner"
KEK	KEK-2283	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Product Owner"
KEK	KEK-2295	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Product Owner"
KEK	KEK-2308	Rol, Expectativa	Independiente, Negociable, Valorable		Rol: "Como Product Owner", Expectativa: "Se realicen pruebas"
KEK	KEK-2309	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	Rol: "Como Product Owner"
KEK	KEK-2310	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	Rol: "Como Product Owner"
KEK	KEK-2312	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner, solicito se realicen las pruebas de estrés"
KEK	KEK-2313				
KEK	KEK-2320	Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2321				
KEK	KEK-1958	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2298	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2311	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2316				
KEK	KEK-2319			Material visual de apoyo (imagen o video)	
KEK	KEK-2322				
KEK	KEK-2325				
KEK	KEK-2329				
KEK	KEK-2330				
KEK	KEK-2338				
KEK	KEK-1957	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como Product Owner"
KEK	KEK-2228	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como Product Owner"
KEK	KEK-2285				
KEK	KEK-2315	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner solicito se realicen las pruebas de regresión "
KEK	KEK-2317	Expectativa, Razón	Independiente, Negociable, Valorable		"llevar a cabo pruebas"

KEK	KEK-2335	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2339	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como Product Owner"
KEK	KEK-2343				
KEK	KEK-2344	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2345				
KEK	KEK-2346	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2347	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2352				
KEK	KEK-2353				
KEK	KEK-1936				
KEK	KEK-1955	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-1959	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2226	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2333	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	"Como Product Owner"
KEK	KEK-2334	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2354	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2355	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como Product Owner"
KEK	KEK-2356	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	
KEK	KEK-2357	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2358	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material documental (guía, manual o lineamiento)	"Como Product Owner"
KEK	KEK-2359				
KEK	KEK-2371				
KEK	KEK-2372	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2373	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2375	Rol, Expectativa	Independiente, Negociable, Valorable, Pequeño		"Como Product Owner"
KEK	KEK-2376				
KEK	KEK-2378	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como Product Owner"
KEK	KEK-2379	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2380	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2383	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	"Como Product Owner"
KEK	KEK-2384				

KEK	KEK-2386	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2387	Rol, Expectativa	Independiente, Negociable, Valorable		"Como Product Owner"
KEK	KEK-2391				
KEK	KEK-2392				
KEK	KEK-2394				
KEK	KEK-2395				
KEK	KEK-2396				
KEK	KEK-2398				
KEK	KEK-2402				
KEK	KEK-2406				
KEK	KEK-2407				
KEK	KEK-2413				
KEK	KEK-2414				
KEK	KEK-2415	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2416	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2417	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2421			Material visual de apoyo (imagen o video)	
KEK	KEK-2422				
KEK	KEK-2423				
KEK	KEK-2424				
KEK	KEK-2425				
KEK	KEK-2408				
KEK	KEK-2273	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2408				
KEK	KEK-2409				
KEK	KEK-2410				
KEK	KEK-2427	Rol, Expectativa	Independiente, Negociable, Valorable		"Yo como Product Owner, Requero que se trabaje en la propuesta"
KEK	KEK-2430				
KEK	KEK-2431				
KEK	KEK-2450				
KEK	KEK-2452				

KEK	KEK-2454				
KEK	KEK-2455				
KEK	KEK-2456				
KEK	KEK-2451	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
KEK	KEK-2273	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como Product Owner"
KEK	KEK-2408				
KEK	KEK-2409				
KEK	KEK-2410				
KEK	KEK-2427				
KEK	KEK-2430				
KEK	KEK-2431				
KEK	KEK-2450				
KEK	KEK-2452				
KEK	KEK-2454				
KEK	KEK-2455				
KEK	KEK-2456				
KEK	KEK-2451	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2462	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2468	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2480				
KEK	KEK-2482				
KEK	KEK-2483				
KEK	KEK-2484				
KEK	KEK-2485	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2486				
KEK	KEK-2477	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2478	Rol, Expectativa	Independiente, Negociable, Valorable		
KEK	KEK-2503				
KEK	KEK-2506	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	
KEK	KEK-2508	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
KEK	KEK-2509	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
KEK	KEK-2511	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		

KEK	KEK-2512	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
KEK	KEK-2513	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"se realicen las pruebas"
KEK	KEK-2519	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	
KEK	KEK-2520				
KEK	KEK-2522				
KEK	KEK-2524				
KEK	KEK-2525				
KEK	KEK-2526				
KEK	KEK-2476	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2530	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	
KEK	KEK-2531	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"se requiere la homologación"
KEK	KEK-2532	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	
KEK	KEK-2533	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2534				
KEK	KEK-2537				
KEK	KEK-2538				
KEK	KEK-2539	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2540				
KEK	KEK-2541				
KEK	KEK-2542				
KEK	KEK-2543				"Establecer el alcance"
KEK	KEK-2544	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		
KEK	KEK-2545	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
KEK	KEK-2547				
KEK	KEK-2548				
KEK	KEK-2549				
KEK	KEK-2550				
IKN	VC-683	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Cómo desarrollador requiero programar"
IKN	VC-758	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-766	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrolladores"
IKN	VC-767	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrolladores"

IKN	VC-768	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-777				
IKN	VC-780	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-781	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-782	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-784				
IKN	VC-783				
IKN	VC-684	Rol, Expectativa	Independiente, Negociable, Valorable		Cómo desarrollador
IKN	C-685	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador"
IKN	VC-689	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador requiero que se valide diseño y funcionalidad del módulo"
IKN	VC-691	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador requiero que se valide diseño y funcionalidad del módulo."
IKN	VC-692	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador requiero que se valide diseño y funcionalidad del módulo."
IKN	VC-772				
IKN	VC-783				
IKN	VC-785	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-786				
IKN	VC-787				
IKN	VC-788	Rol, Expectativa	Independiente, Negociable, Valorable		"Como equipo de desarrollo."
IKN	VC-686	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador requiero que se valide diseño y funcionalidad del módulo."
IKN	VC-804	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video), Material documental (guía, manual o lineamiento)	"Como desarrollador"
IKN	VC-805	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video), Material documental (guía, manual o lineamiento)	"Como desarrollador "
IKN	VC-806	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video), Material documental (guía, manual o lineamiento)	"Como desarrollador"
IKN	VC-808	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video), Material documental (guía, manual o lineamiento)	"Como desarrollador"
IKN	VC-812				
IKN	VC-813				
IKN	VC-815				

IKN	VC-816	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	"Como desarrollador "
IKN	VC-809			Material visual de apoyo (imagen o video)	
IKN	VC-814				
IKN	VC-796	Rol, Expectativa	Independiente, Negociable, Valorable		"Como sistema cliente web"
IKN	VC-797	Rol, Expectativa	Independiente, Negociable, Valorable		
IKN	VC-800	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrollador "
IKN	VC-803				
IKN	VC-687	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador requiero que se valide diseño y funcionalidad del módulo"
IKN	VC-688	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
IKN	VC-778	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como usuario final de cliente web en un dispositivo Android"
IKN	VC-794	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		
IKN	VC-795				
IKN	VC-804	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-805	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador "
IKN	VC-806	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-808	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-812				
IKN	VC-813				
IKN	VC-815				
IKN	VC-816	Rol, Expectativa	Independiente, Negociable, Valorable, Testeable	Material documental (guía, manual o lineamiento)	"Como desarrollador "
IKN	VC-809	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-814				
IKN	VC-761	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-828	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-829	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video), Material documental (guía, manual o lineamiento)	"Como desarrollador"
IKN	VC-838	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-840				
IKN	VC-841				
IKN	VC-842	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador "

IKN	VC-844	Rol, Expectativa	Independiente, Negociable, Valorable	Material documental (guía, manual o lineamiento)	"Como desarrollador"
IKN	VC-845	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-846				
IKN	VC-847	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como desarrollador "
IKN	VC-864	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador "
IKN	VC-848	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-856	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-861	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-862	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-863	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable	Material visual de apoyo (imagen o video)	"Como desarrollador"
IKN	VC-865	Rol, Expectativa	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador "
IKN	VC-870				
IKN	VC-871				
IKN	VC-874				
IKN	VC-875	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador "
IKN	VC-877	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-878	Rol, Expectativa, Razón	Independiente, Negociable, Valorable	Material visual de apoyo (imagen o video)	"Como desarrollador "
IKN	VC-849	Rol, Expectativa	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como desarrollador"
IKN	VC-721	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Cómo desarrollador"
IKN	VC-722	Rol, Expectativa	Independiente, Negociable, Valorable		"Cómo desarrollador"
IKN	VC-887				
IKN	VC-891			Material visual de apoyo (imagen o video)	
IKN	VC-892				
IKN	VC-893			Material visual de apoyo (imagen o video)	
IKN	VC-901				
IKN	VC-906				
IKN	VC-912				
IKN	VC-914	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Cómo desarrollador"
IKN	VC-915			Material documental (guía, manual o lineamiento)	
IKN	VC-923				
IKN	VC-924	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador"

IKN	VC-927	Rol, Expectativa, Razón	Independiente, Negociable, Valorable		"Como desarrollador"
IKN	VC-928				
IKN	VC-929				
IKN	VC-936				
IKN	VC-937	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como desarrollador"
IKN	VC-940				
IKN	VC-941				
IKN	VC-942			Material visual de apoyo (imagen o video)	
IKN	VC-943				
IKN	VC-944				
IKN	VC-945	Rol, Expectativa, Razón	Independiente, Negociable, Valorable, Estimable, Pequeño, Testeable		"Como desarrollador"
IKN	VC-946				
IKN	VC-938			Material visual de apoyo (imagen o video)	
IKN	VC-939			Material visual de apoyo (imagen o video)	
IKN	VC-903			Material visual de apoyo (imagen o video)	
IKN	VC-904			Material visual de apoyo (imagen o video)	
IKN	VC-916			Material visual de apoyo (imagen o video)	
IKN	VC-917				
IKN	VC-938			Material visual de apoyo (imagen o video)	
IKN	VC-939			Material visual de apoyo (imagen o video)	
IKN	VC-949	Rol, Expectativa	Independiente, Negociable, Valorable		"Como desarrollador "
IKN	VC-953				
IKN	VC-957				
IKN	VC-952	Rol, Expectativa	Independiente, Negociable, Valorable	Representación gráfica (diagrama o esquema)	"Como desarrollador"
IKN	VC-960				

8.8 Apéndice H. Resultados de la entrevista

¿Cómo se identifican los requisitos?

Las iniciativas se identifican por medio del *Product Owner* y un comité conformado por personal de negocio y TI. En este comité se intenta “aterrizar” cada iniciativa para ser asignada al *Product Owner* correspondiente. Durante el refinamiento del *backlog* (4 horas por *sprint*) y el *sprint planning* se logran identificar aquellos requisitos no visibles inicialmente o emergentes.

¿Cómo se documentan los requisitos?

Los requisitos se documentan en Jira, Confluence y/o Excel, de acuerdo con el grado de conocimiento que se tenga en ese momento, en Excel se trata como borrador para darle claridad a las ideas, cuando están más claras y entendibles se documentan en Confluence y se procede a documentar en Jira como historia o tarea. Esto puede suceder más usualmente durante el refinamiento del *backlog* y el *sprint planning*.

¿Cómo se estiman los requisitos?

Los requisitos se estiman con base en la incertidumbre que se tenga al respecto y lo complicado que el equipo considera que es el esfuerzo que se debe de aplicar para llevarlo a cabo. Esto se realiza cuando sea necesario durante el refinamiento del *backlog* y el *sprint planning*.

¿Cómo se priorizan los requisitos?

Las iniciativas se priorizan por medio del *Chief Product Owner*, quien se encarga de buscar equilibrar las iniciativas de productos nuevos con las iniciativas de mejora de productos existentes. Si el equipo *Scrum* puede priorizar algún requisito, lo hace, usualmente es cuando conlleva una pronta solución (temas de incidentes). Esto se podría llevar a cabo durante el *Daily Scrum*.

¿Cómo se controla el cambio en los requisitos?

No se controla, se asimila. Actualmente se llega a un mutuo entendimiento con los involucrados de los requisitos, y en caso de surgir un cambio se revisa en conjunto y se procede a comunicar si existiera alguna afectación con respecto a los entregables que ya se habían acordado realizar. Todo esto se realiza, en caso de ser necesario priorizar para cumplir con las expectativas de los involucrados. Esto se podría presentar conforme se va aprendiendo más del producto en desarrollo, durante el *daily Scrum* o el refinamiento del *backlog*.

8.9 Apéndice I. Fichas de síntesis de prácticas de referencia - U.S. Department of Transportation

Identificador:	I-001
Nombre de la práctica:	Desarrollar la descripción general del sistema
Descripción de la Práctica:	<p>Desarrollar una descripción general del sistema que incluya un resumen de sinopsis, describa el contexto en el que será utilizado y liste sus metas, objetivos y restricciones, con el fin de orientar rápidamente a un nuevo lector de los requisitos.</p> <ul style="list-style-type: none"> - Realizarlo temprano en el ciclo de vida de gestión de requisitos y mantener la descripción a un alto nivel para que pueda ser fácil de comprender. - La sinopsis debe incluir el nombre, propósito y capacidades del sistema. - Considerar todo el ciclo de vida del sistema y el contexto en que será utilizado. - Incluir diagramas de contexto en donde se muestre las interacciones del sistema con los elementos externos. Para cada diagrama, incluir una breve descripción de los elementos y las interacciones. - Establecer los objetivos preliminares para que sirvan de guía durante el desarrollo de los requisitos.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

Identificador:	I-002
Nombre de la práctica:	Identificar el alcance del sistema
Descripción de la Práctica:	<p>Desarrollar una definición clara del alcance y límites del sistema en su ambiente. Esto se logra identificando las variables que el sistema monitoreará y controlará.</p> <ul style="list-style-type: none"> - Seleccionar las variables que existen en el contexto. - Seleccionar las variables que el sistema controlará. - Seleccionar las variables que serán monitoreadas por el sistema. - Comprobar que las variables son abstractas y factibles y que no incluyan detalles de implementación. - Definir todas las interfaces físicas del sistema, incluyendo la definición de todas las entradas, mensajes, campos de mensaje y protocolos.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

Identificador:	I-003
Nombre de la práctica:	Desarrollar los conceptos operacionales
Descripción de la Práctica:	<p>Definir como el Sistema interactúa con el ambiente, haciendo uso del concepto de caja negra. Se debe incluir la identificación de las funciones que los usuarios u otros sistemas esperan que el sistema provea, el orden en que estas funciones son invocadas, los valores que pueden proveerse como entradas y la información que el sistema provee como salidas. La utilización de casos de uso es ideal para lograrlo.</p> <ul style="list-style-type: none"> - Documentar el comportamiento ideal del sistema, adicionalmente como una extensión documentar sus fallos y excepciones. - Incluir casos de uso que describan como el sistema es usado en el ambiente operacional. - Utilizar el objetivo de cada caso de uso como su título. - Identificar el usuario principal del caso de uso, las precondiciones y postcondiciones que deben cumplirse. - Asegurar que el caso de uso describa la interacción del usuario con el sistema y otros usuarios. - Relacionar cada paso del caso de uso con cada una de las funciones del sistema. - Consolidar acciones de múltiples casos de uso en uno solo que sea llamado desde distintas locaciones. - Describir situaciones en las que el caso de uso puede fallar en alcanzar su objetivo.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

Identificador:	I-004
Nombre de la práctica:	Identificar los supuestos ambientales
Descripción de la Práctica:	<p>Identificar los supuestos del ambiente para el sistema es esencial para el mantenimiento.</p> <ul style="list-style-type: none"> - Definir el tipo, rango, precisión y unidades supuestas para las variables del sistema. - Proveer una justificación de la causa por la que se hacen los supuestos. - Organizar los supuestos junto al componente externo que impactan.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

Identificador:	I-005
Nombre de la práctica:	Desarrollar la arquitectura funcional
Descripción de la Práctica:	<p>Los requisitos son documentados y organizados como funciones del sistema, con una relación lógica y con la mínima dependencia entre sí. En sistemas complejos, los requisitos son descompuestos en funciones más pequeñas.</p> <ul style="list-style-type: none"> - Organizar las funcionalidades en grupos relacionados que puedan cambiar juntos. - Utilizar diagramas de flujo de datos para representar las funcionalidades y sus dependencias. - Minimizar las dependencias entre funcionalidades asegurando que la información se presenta como conceptos estables y de alto nivel. - Definir el tipo, rango, precisión y unidades de las variables internas. - Organizar las especificaciones de requisitos en múltiples niveles anidando las funciones y dependencias.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia entre requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

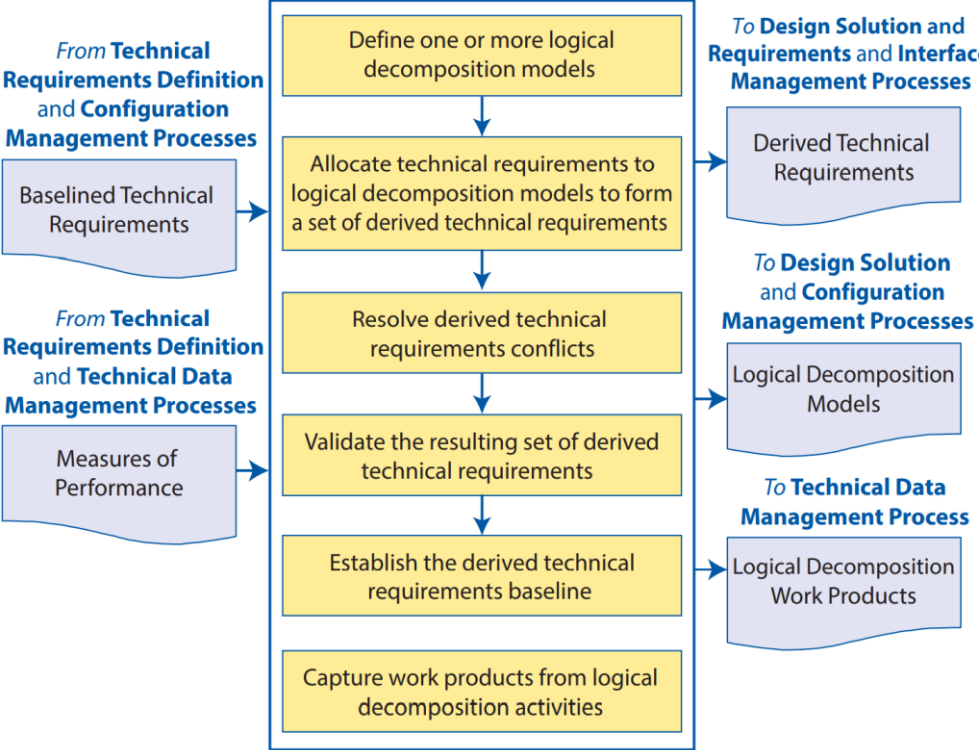
Identificador:	I-006
Nombre de la práctica:	Identificar las restricciones de implementación
Descripción de la Práctica:	<p>Identificar las restricciones del sistema, tales como los requisitos de seguridad, integración o implementación.</p> <ul style="list-style-type: none"> - Modificar la arquitectura funcional si no es capaz de satisfacer las restricciones del sistema. - Revisar la descripción general para reflejar los cambios. - Revisar los conceptos operacionales para reflejar los cambios. - Revisar los casos de uso para identificar las excepciones. - Revisar el alcance del sistema para reflejar los cambios. - Identificar cualquier nuevo supuesto ambiental. - Revisar los diagramas de dependencias. - Revisar los requisitos de alto nivel afectados.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

Identificador:	I-007
Nombre de la práctica:	Desarrollar los requisitos de comportamiento y desempeño
Descripción de la Práctica:	<p>Definir como el sistema cambiará las variables controladas dependiendo del cambio en las variables monitoreadas.</p> <ul style="list-style-type: none"> - Utilizar los nombres de las variables, modos del sistema y las variables internas al redactar los requisitos. - Para cada requisito, especificar los modos del sistema y las condiciones bajo las cuales el requisito aplica, seguido del cambio en la variable afectada. - Asegurar que los requisitos están completos, son consistentes y únicos. - Definir la latencia y tolerancia aceptable para cada variable.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos.
Fuente de información:	U.S. Department of Transportation (2009), <i>Requirements Engineering Management Handbook</i>

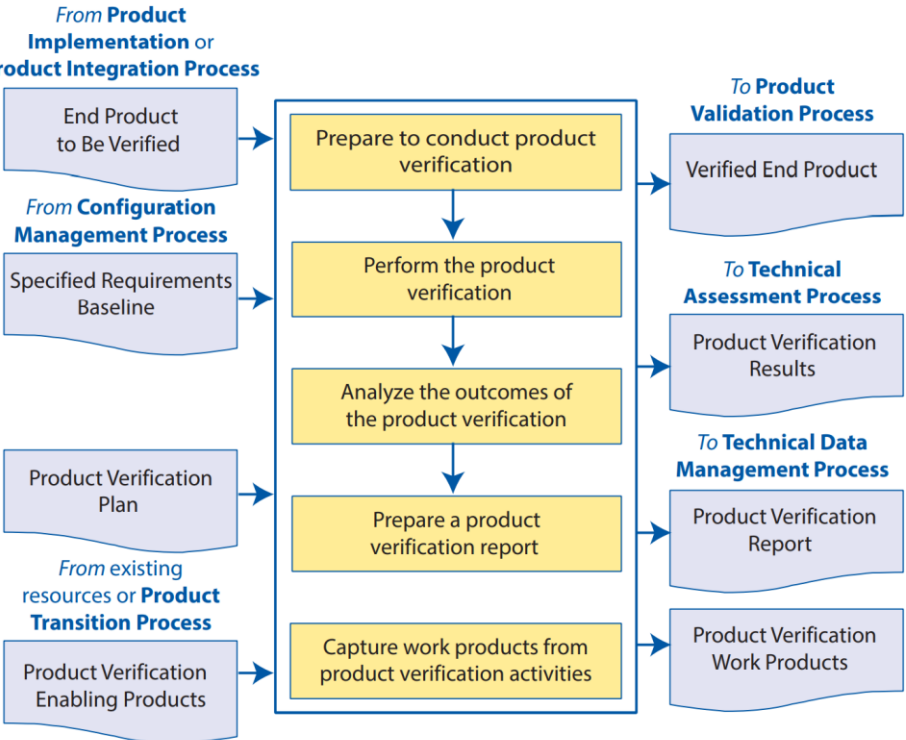
8.10 Apéndice J. Fichas de síntesis de prácticas de referencia - NASA

Identificador:	J-001
Nombre de la práctica:	Definición y gestión de las expectativas de los interesados
Descripción de la Práctica:	<p>Obtener y definir los casos de uso, escenarios y conceptos de operación. Esto debe incluir la identificación de los interesados, el establecimiento de estrategias de soporte, las métricas de efectividad y el compromiso de los interesados.</p> <p>La línea base de expectativas de los interesados será utilizada para validar el producto durante su desarrollo.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-002
Nombre de la práctica:	Definición de requisitos
Descripción de la Práctica:	<p>Transformar la línea base de expectativas de los interesados en requisitos únicos, cuantitativos y medibles. Esto incluye el alcance del problema técnico por resolver, las limitantes que afectan el diseño y los requisitos de desempeño.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

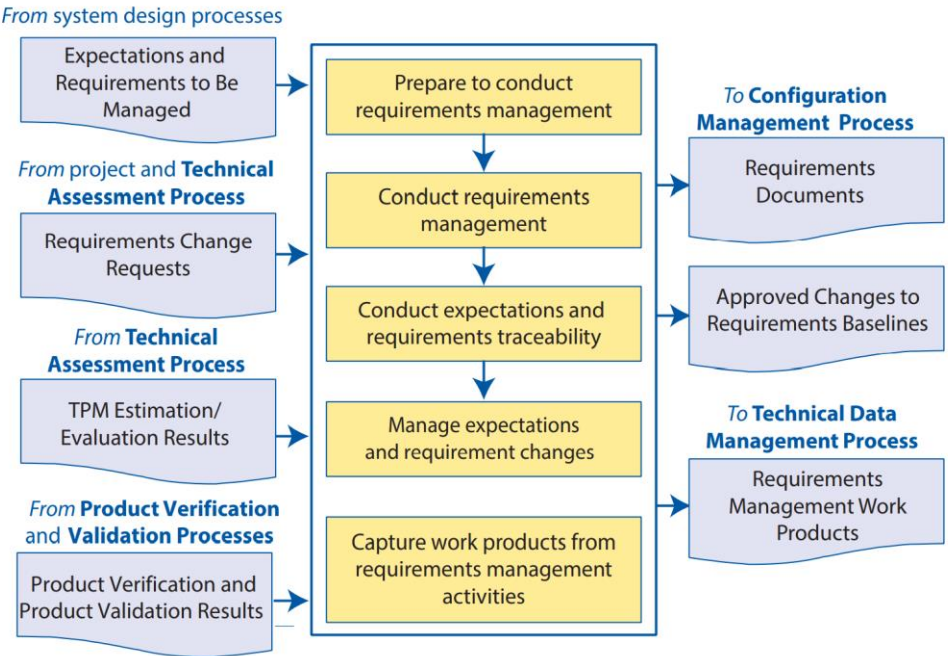
Identificador:	J-003
Nombre de la práctica:	Descomposición lógica
Descripción de la Práctica:	<p>Transformar los requisitos definidos en modelos lógicamente descompuestos con su grupo de requisitos asociados.</p> <p>Esta descomposición puede realizarse por funcionalidad, tiempo, comportamiento, flujo de datos, objetos u otros modelos.</p> <p>Incluye ubicar los requisitos a cada modelo descompuesto, resolviendo los conflictos entre requisitos y definiendo una arquitectura jerárquica de requisitos del sistema.</p>  <pre> graph TD subgraph Inputs A[Baselined Technical Requirements] --> B[Define one or more logical decomposition models] C[Measures of Performance] --> D[Validate the resulting set of derived technical requirements] end B --> E[Allocate technical requirements to logical decomposition models to form a set of derived technical requirements] E --> F[Resolve derived technical requirements conflicts] F --> D D --> G[Establish the derived technical requirements baseline] G --> H[Capture work products from logical decomposition activities] E --> I[Derived Technical Requirements] F --> J[Logical Decomposition Models] G --> K[Logical Decomposition Work Products] subgraph Processes L[From Technical Requirements Definition and Configuration Management Processes] M[From Technical Requirements Definition and Technical Data Management Processes] N[To Design Solution and Requirements and Interface Management Processes] O[To Design Solution and Configuration Management Processes] P[To Technical Data Management Process] end </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

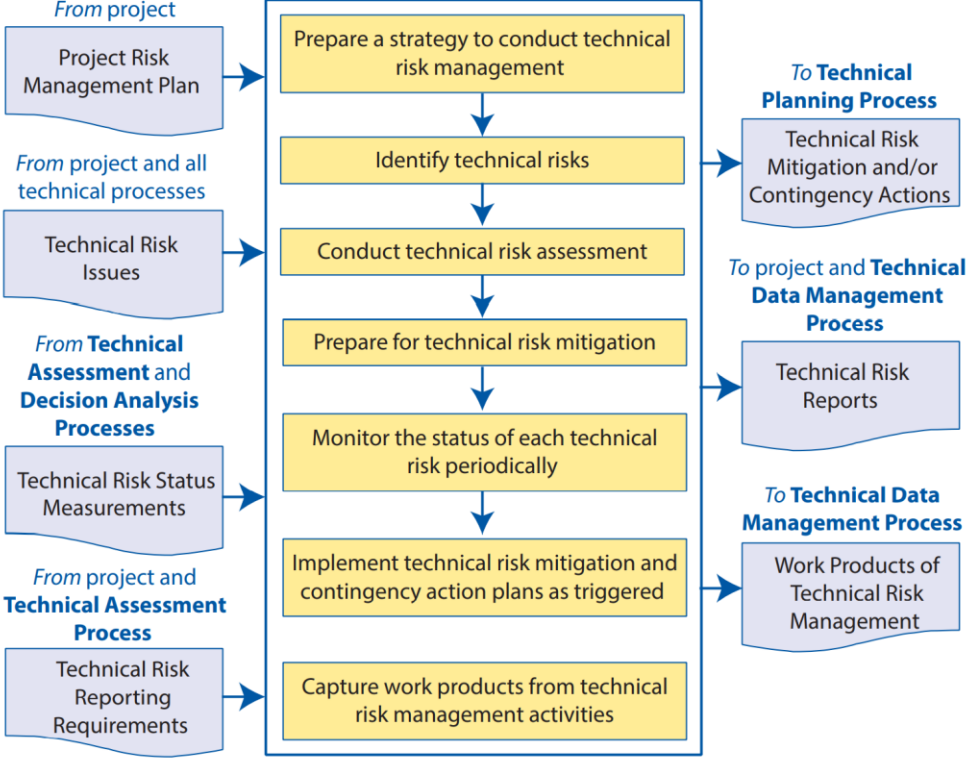
Identificador:	J-004
Nombre de la práctica:	Definición del diseño de solución
Descripción de la Práctica:	<p>Traducir los modelos descompuestos y sus requisitos derivados en uno o más diseños de solución.</p> <p>Mediante el proceso de análisis de decisión, seleccionar la alternativa preferida que satisfaga los requisitos.</p> <p>Desarrollar un paquete completo de datos técnicos para describir la solución seleccionada, que incluye la descripción completa del diseño y las especificaciones de los componentes que serán creados, adquiridos o reutilizados.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-005
Nombre de la práctica:	Verificación del producto
Descripción de la Práctica:	<p>Comprobar que el producto satisfice los requisitos. Esto incluye la identificación de anomalías y el establecimiento de acciones correctivas. Esto debe ser documentados en un reporte de verificación que provea la evidencia de la conformidad del producto con los requisitos.</p> 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia de personas externas. - Cambio de prioridad durante el sprint.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-006
Nombre de la práctica:	Validación del producto
Descripción de la Práctica:	<p>Validar que el producto terminado satisfaga las expectativas de los interesados en el ambiente productivo y asegurar que las anomalías detectadas sean resueltas.</p> <pre> graph TD subgraph Inputs direction TB I1[End Product to Be Validated] I2[Stakeholder Expectation Baseline] I3[Product Validation Plan] I4[Product Validation-Enabling Products] end subgraph Process direction TB P1[Prepare to conduct product validation] P2[Perform the product validation] P3[Analyze the outcomes of the product validation] P4[Prepare a product validation report] P5[Capture the work products from product validation activities] end subgraph Outputs direction TB O1[Validated End Product] O2[Product Validation Results] O3[Product Validation Report] O4[Product Validation Work Products] end I1 --> P1 I2 --> P2 I3 --> P4 I4 --> P5 P1 --> P2 P2 --> P3 P3 --> P4 P4 --> P5 P1 --> O1 P2 --> O2 P4 --> O3 P5 --> O4 </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia de personas externas. - Cambio de prioridad durante el sprint.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-007
Nombre de la práctica:	Planeación técnica
Descripción de la Práctica:	<p>Planificar la ejecución y gestión de cada uno de los procesos técnicos. Identificar, definir y planear los esfuerzos técnicos necesarios para alcanzar los objetivos del proyecto.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia de personas externas. - Cambio de prioridad durante el sprint.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-008
Nombre de la práctica:	Gestión de requisitos
Descripción de la Práctica:	<p>Gestionar los requisitos del producto, incluyendo la trazabilidad bidireccional y el control de cambios para establecer una línea base.</p> <p>Esto incluye, preparar o actualizar la estrategia para la gestión de requisitos, seleccionar una herramienta de gestión apropiada, capacitar a los miembros del equipo técnico en establecer procedimientos de gestión de requisitos, realizar auditorías de trazabilidad de expectativas, controlar los cambios y comunicar las expectativas y los cambios.</p>  <p>The flowchart illustrates the Requirements Management Process. It starts with four input boxes on the left, each with an arrow pointing to a central vertical column of five yellow boxes. The inputs are: 'Expectations and Requirements to Be Managed' (labeled 'From system design processes'), 'Requirements Change Requests' (labeled 'From project and Technical Assessment Process'), 'TPM Estimation/ Evaluation Results' (labeled 'From Technical Assessment Process'), and 'Product Verification and Product Validation Results' (labeled 'From Product Verification and Validation Processes'). The central column contains: 'Prepare to conduct requirements management', 'Conduct requirements management', 'Conduct expectations and requirements traceability', 'Manage expectations and requirement changes', and 'Capture work products from requirements management activities'. Arrows from the central column point to three output boxes on the right: 'Requirements Documents' (labeled 'To Configuration Management Process'), 'Approved Changes to Requirements Baselines', and 'Requirements Management Work Products' (labeled 'To Technical Data Management Process').</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-009
Nombre de la práctica:	Gestión del riesgo técnico
Descripción de la Práctica:	<p>Examinar de forma continua los riesgos de desviación técnica en los planes e identificar potenciales problemas técnicos antes de que ocurran. Planificar, convocar y desempeñar actividades de manejo del riesgo según sea necesario durante el ciclo de vida del proyecto para mitigar el impacto en los objetivos.</p> 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

Identificador:	J-010
Nombre de la práctica:	Documentación de Requisitos
Descripción de la Práctica:	<p>Documentar los requisitos siguiendo las siguientes indicaciones:</p> <ul style="list-style-type: none"> - El requisito debe estar redactado en la forma "el producto ABC debe XYZ", utilizando verbos como hacer, proveer, desempeñar, etc. - El requisito utiliza terminología consistente para referirse al producto. - El requisito establece valores de desempeño, utilizando conceptos como menor que, mayor que o igual a. - El requisito no debe contener aspectos de la solución, solamente del problema que se desea solucionar. - El requisito no debe contener aspectos operativos, no debe representar una actividad. <p>Lista de verificación de un requisito bien redactado:</p> <ul style="list-style-type: none"> - El requisito está gramaticalmente correcto. - El requisito está libre de errores tipográficos, faltas de ortografía y errores de puntuación. - El requisito cumple con la plantilla estándar y reglas de estilo. - El requisito está redactado en positivo. - El requisito es acompañado de un justificante que incluye los supuestos - El requisito está ubicado en la sección correcta del documento. <p>Lista de validación de los requisitos:</p> <ul style="list-style-type: none"> - Claridad: <ul style="list-style-type: none"> o ¿Son los requisitos claros y sin ambigüedad? o ¿Están libre del uso de pronombres indefinidos (este, estos)? o ¿Están libre de términos ambiguos ("según sea apropiado", "etc.", "y/o", "no limitado a")? o ¿Expresan una única sentencia? o ¿La sentencia cuenta con un sujeto y un predicado? - Completitud: <ul style="list-style-type: none"> o ¿Están los requisitos completos? o ¿Se ha pasado por alto algún área de requisitos? o ¿Están todos los supuestos explícitamente establecidos? - Cumplimiento: <ul style="list-style-type: none"> o ¿Están los requisitos en el nivel correcto? o ¿Están libres de especificaciones de implementación? o ¿Están libres de descripciones de operaciones? o ¿Están libres de asignaciones personales o de tareas? - Consistencia: <ul style="list-style-type: none"> o ¿Son los requisitos consistentes y sin contradicciones? o ¿Es la terminología consistente con la terminología del usuario y patrocinador? o ¿Es la terminología consistente en todo el documento? - Trazabilidad: <ul style="list-style-type: none"> o ¿Son todos los requisitos necesarios? o ¿Es cada requisito necesario para satisfacer el requisito padre? o ¿Son todos los requisitos bidireccionalmente trazables hacia los requisitos de alto nivel? o ¿Está cada requisito identificado de forma única?

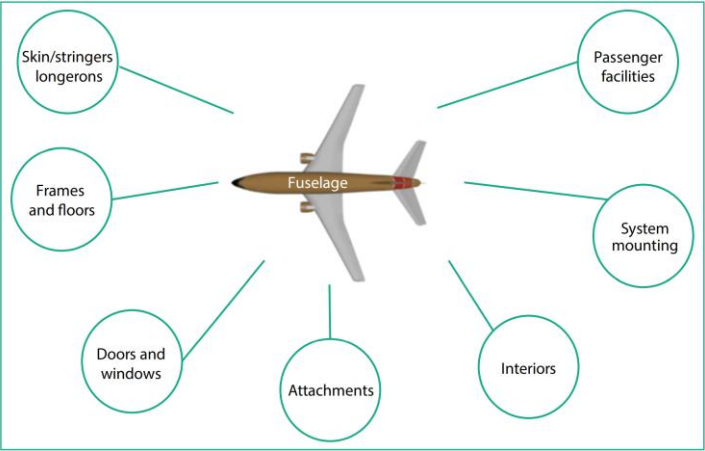
	<ul style="list-style-type: none"> - Correctitud: <ul style="list-style-type: none"> o ¿Son los requisitos correctos? o ¿Los supuestos realizados son correctos? o ¿Son los requisitos técnicamente realizables? - Funcionalidad: <ul style="list-style-type: none"> o ¿Están descritas todas las funcionalidades necesarias para alcanzar los objetivos del sistema? - Desempeño: <ul style="list-style-type: none"> o ¿Están definidas las especificaciones de desempeño? o ¿Es cada requisito de desempeño realista? o ¿Son las tolerancias demasiado ajustadas? - Interfaces: <ul style="list-style-type: none"> o ¿Están claramente definidas todas las interfaces externas? o ¿Están claramente definidas todas las interfaces internas? o ¿Son las interfaces necesarias, suficientes y consistentes entre sí? - Mantenibilidad: <ul style="list-style-type: none"> o ¿Están los requisitos redactados de manera que se minimicen los efectos en cadena debido a cambios? - Confiabilidad: <ul style="list-style-type: none"> o ¿Existen requisitos para la detección, reporte, manejo y recuperación de errores? - Verificabilidad: <ul style="list-style-type: none"> o ¿Puede el sistema ser probado, demostrado, inspeccionado o analizado para verificar que cumple con los requisitos? o ¿Facilitan los requisitos la especificación de casos de prueba? o ¿Están los requisitos libres de términos no verificables? (flexible, fácil, suficiente, seguro, amigable, rápido, portable, pequeño, grande, liviano, robusto o suficiente).
<p>Elementos de la problemática a los que se relaciona o solventa:</p>	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos.
<p>Fuente de información:</p>	<p>NASA (2007), <i>System Engineering Handbook</i></p>

Identificador:	J-011																																																												
Nombre de la práctica:	Matriz de verificación de requisitos																																																												
Descripción de la Práctica:	Herramienta que define la forma en la que se verificarán los requisitos.																																																												
	<table border="1"> <thead> <tr> <th>Requirement No.</th> <th>Document</th> <th>Paragraph</th> <th>Shall Statement</th> <th>Verification Success Criteria</th> <th>Verification Method</th> <th>Facility or Lab</th> <th>Phase*</th> <th>Acceptance Requirement?</th> <th>Preflight Acceptance?</th> <th>Performing Organization</th> <th>Results</th> </tr> </thead> <tbody> <tr> <td><i>Unique identifier or each requirement</i></td> <td><i>Document number the requirement is contained within</i></td> <td><i>Paragraph number of the requirement</i></td> <td><i>Text (within reason) of the requirement, i.e., the "shall"</i></td> <td><i>Success criteria for the requirement</i></td> <td><i>Verification method for the requirement (analysis, inspection, demonstration, test)</i></td> <td><i>Facility or laboratory used to perform the verification and validation.</i></td> <td><i>Phase in which the verification and validation will be performed.</i></td> <td><i>Indicate whether this requirement is also verified during initial acceptance testing of each unit.</i></td> <td><i>Indicate whether this requirement is also verified during any pre-flight or recurring acceptance testing of each unit</i></td> <td><i>Organization responsible for performing the verification</i></td> <td><i>Indicate documents that contain the objective evidence that requirement was satisfied</i></td> </tr> <tr> <td>P-1</td> <td>xxx</td> <td>3.2.1.1 Capability: Support Uplinked Data (LDR)</td> <td>System X shall provide a max. ground-to-station uplink of...</td> <td>1. System X locks to forward link at the min and max data rate tolerances 2. System X locks to the forward link at the min and max operating frequency tolerances</td> <td>Test</td> <td>xxx</td> <td>5</td> <td>Yes</td> <td>No</td> <td>xxx</td> <td>TPS xxxx</td> </tr> <tr> <td>P-i</td> <td>xxx</td> <td>Other paragraphs</td> <td>Other "shalls" in PTRS</td> <td>Other criteria</td> <td>xxx</td> <td>xxx</td> <td>xxx</td> <td>Yes/No</td> <td>Yes/No</td> <td>xxx</td> <td>Memo xxx</td> </tr> <tr> <td>S-i or other unique designator</td> <td>xxxxx (other specs, ICDs, etc.)</td> <td>Other paragraphs</td> <td>Other "shalls" in specs, ICDs, etc.</td> <td>Other criteria</td> <td>xxx</td> <td>xxx</td> <td>xxx</td> <td>Yes/No</td> <td>Yes/No</td> <td>xxx</td> <td>Report xxx</td> </tr> </tbody> </table>	Requirement No.	Document	Paragraph	Shall Statement	Verification Success Criteria	Verification Method	Facility or Lab	Phase*	Acceptance Requirement?	Preflight Acceptance?	Performing Organization	Results	<i>Unique identifier or each requirement</i>	<i>Document number the requirement is contained within</i>	<i>Paragraph number of the requirement</i>	<i>Text (within reason) of the requirement, i.e., the "shall"</i>	<i>Success criteria for the requirement</i>	<i>Verification method for the requirement (analysis, inspection, demonstration, test)</i>	<i>Facility or laboratory used to perform the verification and validation.</i>	<i>Phase in which the verification and validation will be performed.</i>	<i>Indicate whether this requirement is also verified during initial acceptance testing of each unit.</i>	<i>Indicate whether this requirement is also verified during any pre-flight or recurring acceptance testing of each unit</i>	<i>Organization responsible for performing the verification</i>	<i>Indicate documents that contain the objective evidence that requirement was satisfied</i>	P-1	xxx	3.2.1.1 Capability: Support Uplinked Data (LDR)	System X shall provide a max. ground-to-station uplink of...	1. System X locks to forward link at the min and max data rate tolerances 2. System X locks to the forward link at the min and max operating frequency tolerances	Test	xxx	5	Yes	No	xxx	TPS xxxx	P-i	xxx	Other paragraphs	Other "shalls" in PTRS	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Memo xxx	S-i or other unique designator	xxxxx (other specs, ICDs, etc.)	Other paragraphs	Other "shalls" in specs, ICDs, etc.	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Report xxx
	Requirement No.	Document	Paragraph	Shall Statement	Verification Success Criteria	Verification Method	Facility or Lab	Phase*	Acceptance Requirement?	Preflight Acceptance?	Performing Organization	Results																																																	
	<i>Unique identifier or each requirement</i>	<i>Document number the requirement is contained within</i>	<i>Paragraph number of the requirement</i>	<i>Text (within reason) of the requirement, i.e., the "shall"</i>	<i>Success criteria for the requirement</i>	<i>Verification method for the requirement (analysis, inspection, demonstration, test)</i>	<i>Facility or laboratory used to perform the verification and validation.</i>	<i>Phase in which the verification and validation will be performed.</i>	<i>Indicate whether this requirement is also verified during initial acceptance testing of each unit.</i>	<i>Indicate whether this requirement is also verified during any pre-flight or recurring acceptance testing of each unit</i>	<i>Organization responsible for performing the verification</i>	<i>Indicate documents that contain the objective evidence that requirement was satisfied</i>																																																	
	P-1	xxx	3.2.1.1 Capability: Support Uplinked Data (LDR)	System X shall provide a max. ground-to-station uplink of...	1. System X locks to forward link at the min and max data rate tolerances 2. System X locks to the forward link at the min and max operating frequency tolerances	Test	xxx	5	Yes	No	xxx	TPS xxxx																																																	
P-i	xxx	Other paragraphs	Other "shalls" in PTRS	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Memo xxx																																																		
S-i or other unique designator	xxxxx (other specs, ICDs, etc.)	Other paragraphs	Other "shalls" in specs, ICDs, etc.	Other criteria	xxx	xxx	xxx	Yes/No	Yes/No	xxx	Report xxx																																																		
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. 																																																												
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>																																																												

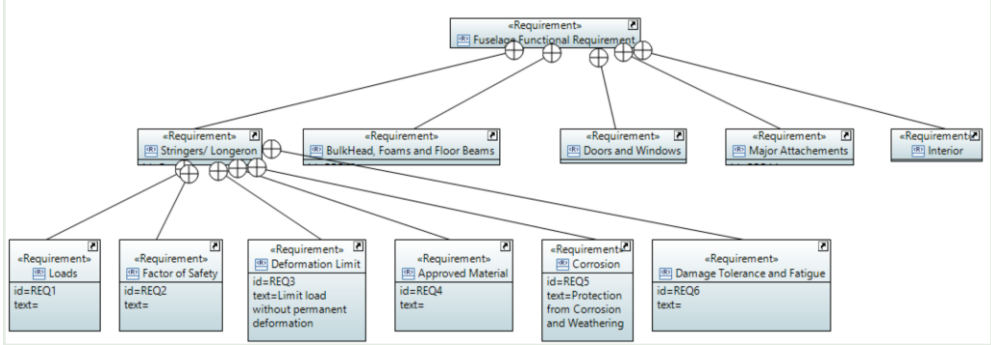
Identificador:	J-012
Nombre de la práctica:	Estructura de desglose del producto
Descripción de la Práctica:	<p>Herramienta que define el alcance del producto y la descomposición de sus elementos.</p> <pre> graph TD CLV[1.3 Crew Launch Vehicle] --> LV[1.3.8 Launch Vehicle] LV --> FS[1.3.8.1 First Stage] LV --> US[1.3.8.2 Upper Stage] LV --> USE[1.3.8.3 Upper Stage Engine] US --> MPS[1.3.8.2.4 MPS] US --> USRCS[1.3.8.2.5 US RCS] US --> FSRCS[1.3.8.2.6 FS RCS] US --> TVCS[1.3.8.2.7 TVCS] US --> Avionics[1.3.8.2.8 Avionics] US --> Software[1.3.8.2.9 Software] US --> ITHW[1.3.8.2.10 Integrated Test H/W] MPS --- MPS_tasks["-1: Integ MPS -2: LH System -3: O2 Fluid Sys. -4: Pressure & Pneumatic Sys. -5: Umbilicals & Disconnect"] USRCS --- USRCS_tasks["-1: Integ RCS -2: Integ Energy Support"] FSRCS --- FSRCS_tasks["-1: Integ RCS"] TVCS --- TVCS_tasks["-1: Integ TVCS -2: Actuator -3: Hydraulic Power -4: APU"] Avionics --- Avionics_tasks["-1: Integ Avionics -2: C&DH System -3: GN&C H/W -4: Radio Frequency System -5: EPS -6: Electrical Integration -7: Develop Flight Instrument -8: Sensor & Instrument System -9: EGSE -10: Integ CLV Avionics System Element Testing -11: Flight Safety System"] Software --- Software_tasks["-1: Integ S/W System -2: Flight S/W"] ITHW --- ITHW_tasks["-1: MPTA -2: GVT -3: STA -4: US for DTF-1 -5: US for VTF-2 -6: US for RRF-3 -7: Struc. Thermal Component Test"] </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
Fuente de información:	NASA (2007), <i>System Engineering Handbook</i>

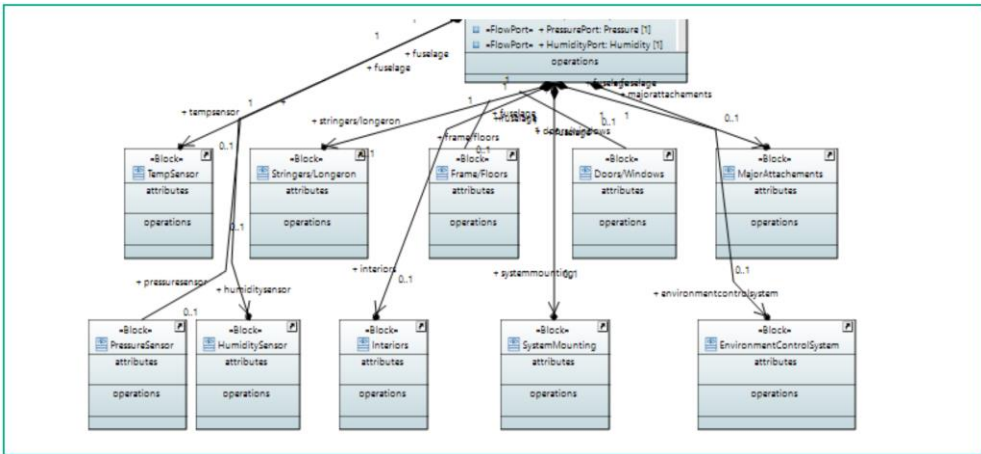
8.11 Apéndice K. Fichas de síntesis de prácticas de referencia – Infosys

Identificador:	K-001
Nombre de la práctica:	Documentación de requisitos
Descripción de la Práctica:	La documentación de requisitos incluye requisitos funcionales, información de materiales, ensamblaje, modelos 2D y 3D, descripción de interfaces, matrices de diseño y planes de pruebas y lanzamiento.
Elementos de la problemática a los que se relaciona o solventa:	- Ambigüedad de los requisitos
Fuente de información:	Infosys (2019), <i>Model-Based System Engineering</i>

Identificador:	K-002
Nombre de la práctica:	Modelado de requisitos
Descripción de la Práctica:	<p>Se captura y procesa información proveniente de distintas fuentes en un único modelo accesible para todos los interesados.</p> <p>Las dependencias entre los diagramas del modelo son establecidas por el usuario, de manera que un cambio en cualquiera de los diagramas se replique a todo el modelo.</p> <p>Cuando un sistema es muy grande, el modelo puede segmentarse en subsistemas de forma que sea posible mantener una relación entre ellos mediante un único modelo.</p> <p>Sistema de sistemas</p> 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos. - Cambios no gestionados en los requisitos.
Fuente de información:	Infosys (2019), <i>Model-Based System Engineering</i>

Identificador:	K-003																									
Nombre de la práctica:	Matriz de requisitos																									
Descripción de la Práctica:	La visualización y relación entre requisitos durante el ciclo de vida del proyecto se realiza mediante una matriz de requisitos.																									
	<table border="1"> <thead> <tr> <th>Fuselage: Major assemblies</th> <th>Requirements</th> <th>Design and simulations</th> <th>Materials and manufacturing</th> <th>Testing and validation</th> <th>Maintenance</th> </tr> </thead> <tbody> <tr> <td>Skin, stringers and longerons</td> <td> <ul style="list-style-type: none"> Load and safety factors Limit load without permanent deformation Approved materials Protection for weathering and corrosion </td> <td> <ul style="list-style-type: none"> Skin - Pressure and shear Longerons bending Stringers axial loads and skin stabilization Skin panel buckling Fail-safe design Least number of joints Minimum weight Fatigue evaluation Concentrated loads </td> <td> <ul style="list-style-type: none"> CFC prepreg Automated fiber placement-CTLM Co-cured and co-bonded </td> <td rowspan="2"> <ul style="list-style-type: none"> Strength tests Coupons Elements Details Subcomponents Components Assembly tests Complete airframe test Stiffness test Ground vibration tests Structural coupling tests Notch filter design </td> <td rowspan="2"> <ul style="list-style-type: none"> Periodic inspection Maintenance as per schedule Replace worn-out, failed or broken components due to operations Define new maintenance schedules </td> </tr> <tr> <td>Frames and floors</td> <td> <ul style="list-style-type: none"> Damage tolerance and fatigue Decompression </td> <td> <ul style="list-style-type: none"> Braiding/resin transfer molding and triaxial-2D braids Fastened to skin </td> <td> <ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges </td> </tr> <tr> <td>Doors and windows</td> <td> <ul style="list-style-type: none"> Safeguards against opening during flight No inadvertent manual operations Emergency exit type, location and number FAR 25.807 Emergency exit access FAR 25.813 </td> <td> <ul style="list-style-type: none"> Interchangeability Wind load while opening Flight load and pressure (ultimate load) Design for ultimate factor </td> <td> <ul style="list-style-type: none"> CFC Aluminum Stops/hinges </td> <td rowspan="2"> <ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges </td> <td rowspan="2"> <ul style="list-style-type: none"> Periodic inspection NDT inspection Replace after designed life </td> </tr> <tr> <td>Major attachments</td> <td> <ul style="list-style-type: none"> Transfer concentrated loads Safe dissipation of load Strength and fatigue requirements </td> <td> <ul style="list-style-type: none"> Selection of type of attachment Major bulk heads Load distribution to shell Structural deflection and instability Discontinuity in the fuselage </td> <td> <ul style="list-style-type: none"> Aluminum Titanium </td> </tr> </tbody> </table>	Fuselage: Major assemblies	Requirements	Design and simulations	Materials and manufacturing	Testing and validation	Maintenance	Skin, stringers and longerons	<ul style="list-style-type: none"> Load and safety factors Limit load without permanent deformation Approved materials Protection for weathering and corrosion 	<ul style="list-style-type: none"> Skin - Pressure and shear Longerons bending Stringers axial loads and skin stabilization Skin panel buckling Fail-safe design Least number of joints Minimum weight Fatigue evaluation Concentrated loads 	<ul style="list-style-type: none"> CFC prepreg Automated fiber placement-CTLM Co-cured and co-bonded 	<ul style="list-style-type: none"> Strength tests Coupons Elements Details Subcomponents Components Assembly tests Complete airframe test Stiffness test Ground vibration tests Structural coupling tests Notch filter design 	<ul style="list-style-type: none"> Periodic inspection Maintenance as per schedule Replace worn-out, failed or broken components due to operations Define new maintenance schedules 	Frames and floors	<ul style="list-style-type: none"> Damage tolerance and fatigue Decompression 	<ul style="list-style-type: none"> Braiding/resin transfer molding and triaxial-2D braids Fastened to skin 	<ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges 	Doors and windows	<ul style="list-style-type: none"> Safeguards against opening during flight No inadvertent manual operations Emergency exit type, location and number FAR 25.807 Emergency exit access FAR 25.813 	<ul style="list-style-type: none"> Interchangeability Wind load while opening Flight load and pressure (ultimate load) Design for ultimate factor 	<ul style="list-style-type: none"> CFC Aluminum Stops/hinges 	<ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges 	<ul style="list-style-type: none"> Periodic inspection NDT inspection Replace after designed life 	Major attachments	<ul style="list-style-type: none"> Transfer concentrated loads Safe dissipation of load Strength and fatigue requirements 	<ul style="list-style-type: none"> Selection of type of attachment Major bulk heads Load distribution to shell Structural deflection and instability Discontinuity in the fuselage
Fuselage: Major assemblies	Requirements	Design and simulations	Materials and manufacturing	Testing and validation	Maintenance																					
Skin, stringers and longerons	<ul style="list-style-type: none"> Load and safety factors Limit load without permanent deformation Approved materials Protection for weathering and corrosion 	<ul style="list-style-type: none"> Skin - Pressure and shear Longerons bending Stringers axial loads and skin stabilization Skin panel buckling Fail-safe design Least number of joints Minimum weight Fatigue evaluation Concentrated loads 	<ul style="list-style-type: none"> CFC prepreg Automated fiber placement-CTLM Co-cured and co-bonded 	<ul style="list-style-type: none"> Strength tests Coupons Elements Details Subcomponents Components Assembly tests Complete airframe test Stiffness test Ground vibration tests Structural coupling tests Notch filter design 	<ul style="list-style-type: none"> Periodic inspection Maintenance as per schedule Replace worn-out, failed or broken components due to operations Define new maintenance schedules 																					
Frames and floors	<ul style="list-style-type: none"> Damage tolerance and fatigue Decompression 	<ul style="list-style-type: none"> Braiding/resin transfer molding and triaxial-2D braids Fastened to skin 	<ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges 																							
Doors and windows	<ul style="list-style-type: none"> Safeguards against opening during flight No inadvertent manual operations Emergency exit type, location and number FAR 25.807 Emergency exit access FAR 25.813 	<ul style="list-style-type: none"> Interchangeability Wind load while opening Flight load and pressure (ultimate load) Design for ultimate factor 	<ul style="list-style-type: none"> CFC Aluminum Stops/hinges 	<ul style="list-style-type: none"> Periodic inspection as per schedule Replacement of movable parts and hinges 	<ul style="list-style-type: none"> Periodic inspection NDT inspection Replace after designed life 																					
Major attachments	<ul style="list-style-type: none"> Transfer concentrated loads Safe dissipation of load Strength and fatigue requirements 	<ul style="list-style-type: none"> Selection of type of attachment Major bulk heads Load distribution to shell Structural deflection and instability Discontinuity in the fuselage 	<ul style="list-style-type: none"> Aluminum Titanium 																							
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> Dependencia entre requisitos. Cambios no gestionados en los requisitos. 																									
Fuente de información:	Infosys (2019), <i>Model-Based System Engineering</i>																									

Identificador:	K-004
Nombre de la práctica:	Descomposición de requisitos
Descripción de la Práctica:	<p>La documentación y descomposición de los requisitos de forma jerárquica permite realizar una mejor trazabilidad.</p> 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Infosys (2019), <i>Model-Based System Engineering</i>

Identificador:	K-005
Nombre de la práctica:	Diagrama de estructura
Descripción de la Práctica:	<p>El diagrama de estructura es una representación de los componentes de hardware y software que conforman el producto final.</p> 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Infosys (2019), <i>Model-Based System Engineering</i>

Identificador:	K-006
Nombre de la práctica:	Diagrama de comportamiento
Descripción de la Práctica:	<p>El diagrama de comportamiento simula el comportamiento del sistema para satisfacer los requisitos. Las acciones del sistema son ejecutadas basadas en la transformación de las entradas en salidas.</p> <pre> graph TD Start((Airplane started)) --> Taxi((Taxi)) Taxi -- "Altitude >= 0 Speed > 0 km/h" --> Ascent((Ascent)) Ascent -- "Altitude >= 35000 feet Speed > 850 km/h" --> InFlight((In-flight)) InFlight -- "Altitude < 35000 feet Speed < 950 km/h" --> Descent((Descent)) Descent -- "Altitude >= 0 Speed < 50 km/h" --> Taxi Ascent -- "Pressure threshold" --> Taxi </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Infosys (2019), Model-Based System Engineering

Identificador:	K-007
Nombre de la práctica:	Diagrama paramétrico
Descripción de la Práctica:	<p>El diagrama paramétrico captura la relación entre las propiedades del sistema para permitir los análisis de desempeño, fiabilidad y estrés.</p> <pre> graph TD A[Limit Load for fuselage components (from Loads & Aero Elasticity Group)] --> B[Ultimate Load on each component=Limit Load X Load Factor (based on FAR 25)] B --> C[Actual Stress (P_actual)] B --> D[Allowable Stress (P_allowable)] C --> E["Margin of Safety (MoS) = (P_allowable / P_actual) - 1 > 0.0"] D --> E </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Infosys (2019), Model-Based System Engineering

8.12 Apéndice L. Fichas de síntesis de buenas prácticas

Identificador:	L-001
Nombre de la práctica:	Evaluación de la necesidad
Descripción de la Práctica:	<p>Prácticas para la evaluación de la necesidad:</p> <ul style="list-style-type: none"> - Análisis FODA: un análisis de fortalezas, debilidades, oportunidades y amenazas ayuda a obtener una perspectiva de alto nivel sobre una oportunidad de negocio o problemática a resolver y comparar opciones durante la gestión del proyecto. - Análisis de decisión: provee las bases para realizar una toma de decisiones estructurada y analítica. Tablas y árboles de decisión permiten representar series de decisiones y sus resultados. Los árboles de decisión funcionan mejor cuando existen selecciones binarias y las tablas de decisión cuando existen más de dos opciones. - Análisis de brechas: compara el nivel actual de las capacidades organizacionales contra un estado futuro deseado. - Benchmarking: provee información al respecto de cómo otras organizaciones responden ante desafíos similares y permite comparar las prácticas actuales de la organización a aquellas mejores prácticas ejecutadas por organizaciones similares.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas.
Fuente de información:	Project Management Institute (2016). Requirement Management: A Practice Guide.

Identificador:	L-002
Nombre de la práctica:	Obtención de requisitos
Descripción de la Práctica:	<p>Prácticas para la obtención de requisitos:</p> <ul style="list-style-type: none"> - Entrevista: permite obtener información por parte de los interesados mediante la realización de preguntas y la documentación de las respuestas. Durante esta actividad, los participantes del proyecto identifican funcionalidades que deben estar presentes en el producto final, servicio o resultado. Las preguntas pueden estar definidas previo al inicio de la entrevista o pueden ser realizadas de una manera interactiva con forme se obtienen respuestas a las preguntas anteriores. - Sesiones de trabajo: se reúnen los interesados y los equipos multidisciplinares en una sesión estructurada y enfocada en la identificación de los requisitos, la reconciliación de las diferencias y al alcance de concesos. - Grupos de enfoque: se reúnen a participantes calificados en un grupo para compartir sus pensamientos y expectativas al respecto del producto, servicio o resultado. Esta técnica provee retroalimentación cualitativa que puede ser examinada durante el análisis de los requisitos. - Lluvia de ideas: los participantes generar múltiples ideas relacionadas a un tema en particular. Se analizan las diferentes perspectivas del problema o solución y se construye sobre las ideas de los demás.

	<ul style="list-style-type: none"> - Cuestionarios o encuestas: obtienen información de un gran grupo de usuarios de forma rápida. Se componen de una serie de preguntas destinadas a obtener datos. Estos datos son analizados para extraer requisitos relevantes. Si las preguntas cuentan con respuestas predefinidas, se conocen como preguntas cerradas. Por su parte, las preguntas abiertas permiten generar respuestas amplias. - Análisis documental: se inspeccionan documento como glosarios de términos, planes estratégicos y de negocios, flujos de procesos, registros de errores, regulaciones, políticas y procedimiento para extraer información relevante. - Análisis de interfaz: examinar las interacciones del sistema con los usuarios, proceso y otros componentes del sistema permite identificar los requisitos. Esta técnica también permite identificar interesados adicionales que podrían verse impactados por cambios en las interfaces del sistema. - Prototipo: provee un modelo funcional del producto esperado previo a su desarrollo con la finalidad de obtener retroalimentación temprana de los requisitos. Esta técnica permite refinar los requisitos de manera progresiva, ya que brinda la oportunidad a los interesados de probar, experimentar y brindar retroalimentación. - Observación: provee información mediante la observación de las personas en su ambiente durante la ejecución de sus tareas.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
Fuente de información:	Project Management Institute (2016). Requirement Management: A Practice Guide.

Identificador:	L-003
Nombre de la práctica:	Análisis de requisitos
Descripción de la Práctica:	<p>Prácticas para el análisis de requisitos:</p> <ul style="list-style-type: none"> - Gestión y priorización del <i>backlog</i>: Los elementos que se encuentran en la parte superior del <i>backlog</i> deben contener suficientes detalles para su realización, a este trabajo se le conoce como <i>grooming the backlog</i>. Para priorizar cuales elementos tendrán prioridad se utilizan las siguientes técnicas: <ul style="list-style-type: none"> o MoSCoW: cada elemento se clasifica según el siguiente criterio: <ul style="list-style-type: none"> ▪ <i>Must have</i>: El requisito es fundamental para el éxito de la solución. ▪ <i>Should have</i>: El requisito es importante pero no compromete el éxito. ▪ <i>Could have</i>: El requisito puede ser eliminado sin impacto. ▪ <i>Won't have</i>: El requisito no será entregado durante la iteración actual. o Votar: los participantes asignan votos a los requisitos que consideran más importantes. Los votos resultantes permiten comparar la importancia de uno sobre otro para establecer un orden de prioridad.

	<ul style="list-style-type: none"> ○ Basado en el tiempo: se analiza la capacidad de trabajo a ser ejecutado en un específico período de tiempo y se definen cuales requisitos podrán ser entregados en ese tiempo. - Modelado: Existen múltiples tipos de modelos que permiten analizar los requisitos. El tipo de modelo será determinado según la información que se desea compartir y la audiencia que la recibirá. <ul style="list-style-type: none"> ○ Modelo de alcance: identifica los límites del sistema, expresa sus funcionalidades, capacidades y límites. <ul style="list-style-type: none"> ▪ Diagrama de contexto: representa el alcance del sistema mediante sus interacciones con las personas y otros sistemas. Provee las entradas, salidas y el flujo de información de cada elemento. ▪ Mapa de ecosistema: ilustra los componentes del sistema, incluyendo las relaciones existentes entre las personas, los datos, el <i>hardware</i> y el <i>software</i>. Muestra las interacciones directas e indirectas. ▪ Modelos de objetivos de negocio: organiza las metas, objetivos de negocio, y las funcionalidades de alto nivel. Provee una estructura para especificar los requisitos de negocio que se alinean a los objetivos de alto nivel. ▪ Modelo de funcionalidad: muestra la organización de los grupos de requisitos según su funcionalidad y las relaciones entre ellos. Esto permite identificar requisitos duplicados o faltantes. ▪ Diagrama de caso de uso: es una representación gráfica de los casos de uso del sistema desde el punto de vista de un rol de usuario. Los casos de uso no muestran los requisitos, pero permiten organizarlos para su análisis. ○ Modelo funcional: organiza la lógica y dependencias de las relaciones entre las funciones para representar lo que debe ser desarrollado. Pueden representar la línea base o el estado futuro de un proyecto, producto o proceso y permiten descubrir funciones que son difíciles de identificar. ○ Modelo de procesos: describe las interacciones del usuario para un proceso particular. Permite analizar los requisitos incluidos en el flujo del proceso, caso de uso e historias de usuario. <ul style="list-style-type: none"> ▪ Flujo de proceso: representa las actividades desarrolladas, incluyendo la secuencia de pasos, decisiones y roles responsables de ejecutar cada paso. ▪ Caso de uso: describe un comportamiento del sistema desde la perspectiva del usuario y provee una visión de alto nivel de la funcionalidad. Facilita identificar los requisitos mediante la clarificación de los objetivos que pretende el interesado al interactuar con el sistema. ▪ Historia de usuario: es una sentencia que captura una nueva funcionalidad del sistema desde el punto de vista del usuario. Las historias de usuario pueden ser usados para gestionar, priorizar, dar seguimiento y colocar funcionalidades a las iteraciones. ○ Modelo de reglas: documenta políticas de negocio, reglas de negocio y decisiones que impactan la solución.
--	--

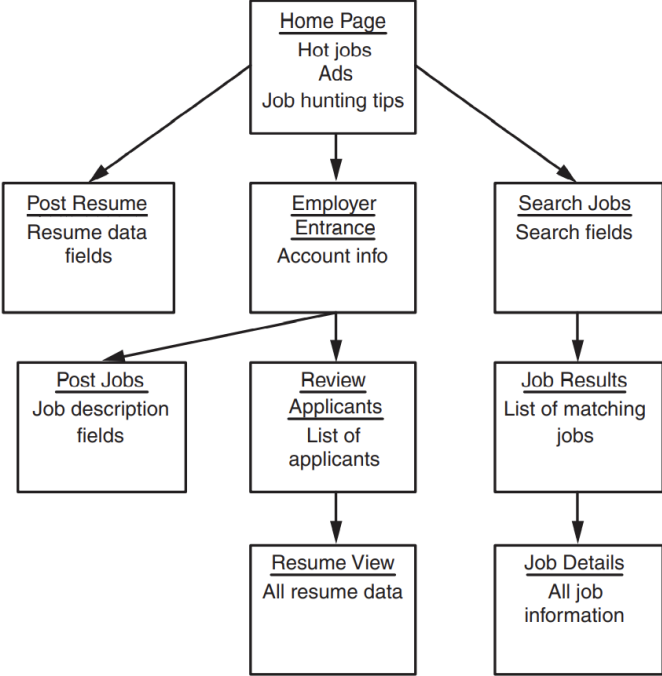
	<ul style="list-style-type: none"> ▪ Catálogo de reglas de negocio: es un repositorio de atributos relacionados en el cual se definen guías y estándares que influyen en el comportamiento de la solución. ▪ Tabla o árbol de decisión: documenta una serie de decisiones y sus resultados. Estos modelos son utilizados para representar reglas complejas de negocio, incluyendo posibles condiciones y acciones. ○ Modelo de datos: describe las necesidades específicas de información de un sistema. Mostrando las relaciones entre los datos y los procesos, este modelo provee detalles para extraer requisitos y sus reglas de negocio relacionadas. <ul style="list-style-type: none"> ▪ Diagrama de relación de entidad: muestra los objetos de datos de negocio y sus relaciones. ▪ Diagrama de flujo de datos: retrata el movimiento de los datos a través del sistema y como son manipulados por los procesos. Documenta donde se almacena la información e identifica entradas y salidas de los procesos. ▪ Diccionario de datos: provee una descripción de los datos, atributos y propiedades que definen los objetos de datos. ▪ Diagrama de tabla de estado: ilustra la transición de los datos en cada uno de sus estados. ○ Modelo de interfaz: documenta las relaciones e interacciones entre sistemas y/o usuario con la solución. <ul style="list-style-type: none"> ▪ Tabla de reporte: describe los requisitos necesarios para desarrollar y mostrar información en un solo reporte. Este define como los datos son manipulados y mostrados al usuario. ▪ Tabla de interfaz de sistema: documenta el flujo de comunicación y transferencia de datos entre la fuente y los sistemas objetivo. Incluye objetos de datos específicos, el volumen de datos transferidos y la frecuencia de la transmisión. ▪ Flujo de interfaz de usuario: describe la comunicación entre un usuario y un sistema. Representa como los usuarios manipulan el sistema para completar una tarea. ▪ Modelo <i>DAR/Wireframe</i>: documenta como un sistema muestra los datos y como el sistema responde a las acciones iniciadas por el usuario. Hace uso de <i>mockups</i> de pantalla para la interfaz de usuario. ▪ Diagrama N2: identifica y representa las interfaces entre los elementos del sistema en un formato tabular. Ayuda a identificar y dar seguimiento a los requisitos que afectan más de un elemento del sistema.
<p>Elementos de la problemática a los que se relaciona o solventa:</p>	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
<p>Fuente de información:</p>	<p>Project Management Institute (2016). Requirement Management: A Practice Guide.</p>

Identificador:	L-004
Nombre de la práctica:	Monitoreo y control de requisitos
Descripción de la Práctica:	<p>Prácticas para el monitoreo y control de requisitos:</p> <ul style="list-style-type: none"> - Análisis de dependencias: permite identificar relaciones de dependencia entre requisitos. El grupo de requisitos es documentado en una matriz de trazabilidad, agrupando los requisitos dependientes. Un árbol de trazabilidad permite visualizar estas dependencias. - Análisis de impacto: evalúa como un cambio en un requisito afecta al resto, al producto y al proyecto. Pretende identificar el riesgo asociado al cambio, el trabajo requerido para incorporar el cambio y sus implicaciones de costos y cronograma. - Matriz de trazabilidad: es un cuadro que relaciona los requisitos desde su origen hasta el entregable que los satisface. Al relacionar cada requisito con los objetivos de negocio y del proyecto se asegura que generen el valor de negocio esperado. Además, permite gestionar el cambio de forma estructurada y gestionar el alcance del producto. - Pizarra de control de cambios: consiste en un grupo de interesados responsables por revisar, evaluar, aprobar, retrasar, rechazar y comunicar cambios en el proyecto. Esta práctica se recomienda para proyecto fuertemente regulados o que cuenten con un gran número de componentes, interfaces, riesgos e interesados. Una vez ha sido creada la línea base de requisitos, cualquier cambio propuesto debe ser gestionado por la pizarra de control de cambios.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia de personas externas. - Dependencia entre requisitos. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	Project Management Institute (2016). Requirement Management: A Practice Guide.

Identificador:	L-005
Nombre de la práctica:	Evaluación de la solución
Descripción de la Práctica:	<p>Prácticas para la evaluación de la solución:</p> <ul style="list-style-type: none"> - Solicitar opiniones: solicitar su opinión a los interesados, usuarios finales o personal con experto puede ayudar a validar si el producto, servicio o resultado se comporta como se espera. Para esto se suelen utilizar técnicas como revisiones, grupos de enfoque, encuestas, lluvia de ideas o listas de verificación. - Pruebas: pruebas de exploración o de aceptación de usuario para validar que la solución cumple con los criterios de aceptación. - Demostraciones: operar la solución para probar o mostrar que cumple con las funciones esperadas.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	Project Management Institute (2016). Requirement Management: A Practice Guide.

Identificador:	L-006
Nombre de la práctica:	Cierre del proyecto o fase
Descripción de la Práctica:	<p>Prácticas para el cierre del proyecto o fase:</p> <ul style="list-style-type: none"> - Las técnicas utilizadas previamente durante el análisis pueden ser de utilidad durante el cierre para determinar las limitaciones de la solución implementada. - Reuniones: se discuten y documentan temas relacionados al cierre del proyecto o fase. Estas reuniones pueden ser cara a cara, virtuales, formales o informales y normalmente incluyen la transferencia del conocimiento y la documentación de lecciones aprendidas. Estas sesiones permiten comprender el éxito y desafío de la iniciativa e identificar las áreas de mejora.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	Project Management Institute (2016). Requirement Management: A Practice Guide.







Identificador:	L-007
Nombre de la práctica:	Definición de roles de usuario
Descripción de la Práctica:	<p>Proceso mediante el cual son definidos los roles de usuario del sistema.</p> <ul style="list-style-type: none"> - Identificar los roles de usuario mediante una lluvia de ideas. El cliente y los desarrolladores realizan una sesión conjunta en la cual proponen nombre para roles de usuario y los colocan en una pizarra visible para todos. Cada uno de los roles debe representar a un individuo, no a un colectivo. - Organizar los roles de usuario. Los participantes agrupan los roles relacionados y eliminan los repetidos de la pizarra sobreponiéndolos. - Consolidar los roles de usuario. Unificar los roles sobrepuestos en la pizarra asegurando que realmente representan al mismo usuario. Además, remover cualquier rol de usuario que no sea importante para el éxito del sistema. - Refinar los roles. Definir atributos para cada rol que facilite su identificación y lo diferencie del resto.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas.
Fuente de información:	Cohn, M. (2004). <i>User Stories Applied for Agile Software Development</i> .

Identificador:	L-008
Nombre de la práctica:	Obtención de historias de usuario
Descripción de la Práctica:	<p>Proceso para obtener historias de usuario que debe ser ejecutado al menos una vez al inicio de cada planificación de entrega.</p> <ul style="list-style-type: none"> - Se realiza una sesión en la cual participa el cliente, los usuarios, los desarrolladores y otros interesados que puedan colaborar en la escritura de las historias. - Se elige el rol de usuario que se considere más relevante. - Los participantes escriben todas las funcionalidades del sistema que consideren necesarias para el rol de usuario elegido. - Se organizan las historias en orden de importancia y se eliminan las historias repetidas. - Se dibuja un prototipo de baja fidelidad que muestre los componentes del sistema y sus interacciones para el rol de usuario elegido, basado en las historias sugeridas. - Repetir el proceso para cada rol de usuario.  <pre> graph TD HP["Home Page Hot jobs Ads Job hunting tips"] --> PR["Post Resume Resume data fields"] HP --> EE["Employer Entrance Account info"] HP --> SJ["Search Jobs Search fields"] EE --> PJ["Post Jobs Job description fields"] EE --> RA["Review Applicants List of applicants"] RA --> RV["Resume View All resume data"] SJ --> JR["Job Results List of matching jobs"] JR --> JD["Job Details All job information"] </pre>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
Fuente de información:	Cohn, M. (2004). <i>User Stories Applied for Agile Software Development</i> .

Identificador:	L-009
Nombre de la práctica:	Formulación de historias de usuario
Descripción de la Práctica:	<p>A continuación, se detallan algunas buenas prácticas para la formulación de historias:</p> <ul style="list-style-type: none"> - En proyectos grandes en donde es complejo identificar a todos los roles de usuario, se debe iniciar con la formulación de historias que representen los objetivos que se desean alcanzar al utilizar el sistema. - Cuando las historias son muy grandes para poder realizarlas en una iteración es necesario dividir las historias en historias más pequeñas. Las nuevas historias deben proveer valor al usuario final, debe evitarse la división de historias basada en aspectos técnicos. Una historia bien formulada es aquella que al ser completada permite al usuario contar con una nueva funcionalidad. - Adjuntar a la historia cualquier restricción, supuesto o dependencia. - Priorizar la división de las historias según el orden de las entregas a realizar. - Utilizar el rol de usuario en la formulación de cada historia de usuario. - Utilizar la voz activa. - Idealmente el cliente es quien redacta la historia de usuario. - No enumerar las historias de usuario. - Evitar cualquier especificación relacionada a la solución, como por ejemplo aspectos de la interfaz de usuario. - No utilizar historias de usuario siempre. Si hay alguna funcionalidad que es mejor descrita mediante imágenes o algún otro formato, no intentar forzar el uso de historia.
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos.
Fuente de información:	Cohn, M. (2004). <i>User Stories Applied for Agile Software Development</i> .


Identificador:	L-010																														
Nombre de la práctica:	Estimación de historias de usuario																														
Descripción de la Práctica:	<p>A continuación, se detallan algunas buenas prácticas para la estimación de historias:</p> <ul style="list-style-type: none"> - Puntos de historia. Se define un punto de historia como un día de trabajo ideal, en el cual no existe ningún tipo de interrupción. - Estimar como equipo. Debido a que no hay certeza de quien ejecutará cada tarea, el estimar como equipo es la forma más precisa de hacerlo. Solo el equipo de trabajo participa en la estimación. - Técnica de estimación. Se reúne el equipo de desarrollo con el cliente, se selecciona una historia de usuario al azar y se realizan todas las preguntas que aclaren su alcance. Cada miembro del equipo escribe en un papel el valor estimado sin mostrarlo a los demás, una vez todos han finalizado se muestran los estimados y si difieren, se aclaran las razones. Una vez el equipo llegue a un consenso del valor estimado se continúa con la siguiente historia. - Triangular. Una vez han sido estimadas algunas historias es necesario triangular los estimados comparando los resultados entre si y analizando si los estimados son coherentes. - Pronosticar. Utilizando la cantidad total de puntos de historia de usuario completados en una iteración es posible realizar un pronóstico del trabajo a completarse en futuras iteraciones. <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="width: 15%;">1</th> <th style="width: 15%;">2</th> <th style="width: 15%;">3</th> <th style="width: 15%;">5</th> <th style="width: 15%;">8</th> <th style="width: 15%;">13</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> </tr> <tr> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td></td> <td style="text-align: center;">A user can...</td> </tr> <tr> <td style="text-align: center;">A user can...</td> <td></td> <td style="text-align: center;">A user can...</td> <td style="text-align: center;">A user can...</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">A user can...</td> <td></td> <td></td> <td style="text-align: center;">A user can...</td> <td></td> <td></td> </tr> </tbody> </table>	1	2	3	5	8	13	A user can...	A user can...	A user can...	A user can...	A user can...	A user can...	A user can...	A user can...	A user can...	A user can...		A user can...	A user can...		A user can...	A user can...			A user can...			A user can...		
1	2	3	5	8	13																										
A user can...	A user can...	A user can...	A user can...	A user can...	A user can...																										
A user can...	A user can...	A user can...	A user can...		A user can...																										
A user can...		A user can...	A user can...																												
A user can...			A user can...																												
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. 																														
Fuente de información:	Cohn, M. (2004). <i>User Stories Applied for Agile Software Development</i> .																														

Identificador:	L-011																																							
Nombre de la práctica:	Documento de requisitos del producto (PRD)																																							
Descripción de la Práctica:	<p>Este documento define los requisitos de un producto concreto, incluidos el propósito, las funciones, la funcionalidad y el comportamiento de este.</p> <p>El PRD unifica la documentación de los requisitos de un producto, permite compartirlo con los interesados y obtener su aprobación.</p> <ul style="list-style-type: none"> - Definir las especificaciones del proyecto. Participantes: ¿Quién participa? Incluye el propietario del producto, el equipo, las partes interesadas. Estado: ¿Cuál es el estado actual del programa? Cumple plazos, en peligro, retrasado, aplazado, etc. Objetivo de publicación: ¿Cuándo está planificado su lanzamiento? - Metas del equipo y objetivos empresariales. - Contexto y enfoque estratégico. ¿Por qué hacemos esto? ¿Cómo encaja en los objetivos generales de la empresa? - Suposiciones - Historias de Usuario. Especificar o proporcionar enlaces a las historias de usuario relevantes. Incluyendo enlaces a entrevistas a los clientes y capturas de pantalla. - Diseño e interacción con los usuarios. Incluir un enlace a los estudios de diseño y esquemas de la página. - Preguntas. Crear una tabla de lo que hay que decidir o investigar para realizar un seguimiento de estos asuntos. - Limitaciones del alcance Señalar claramente las tareas que no hay que hacer. <div data-bbox="558 974 1442 1869" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <h3 style="margin: 0;">Mobile Web Requirements</h3> <p style="font-size: 0.8em; margin: 0;">Created by Mitch Davis, last modified just a moment ago</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="width: 20%;">Target release</td> <td>1.0</td> </tr> <tr> <td>Epic</td> <td>MDT-18 - Mobile optimized web app TO DO</td> </tr> <tr> <td>Document status</td> <td>DRAFT</td> </tr> <tr> <td>Document owner</td> <td>@ Mitch Davis</td> </tr> <tr> <td>Designer</td> <td>@ Cassie Owens</td> </tr> <tr> <td>Developers</td> <td>@ Harvey Jennings</td> </tr> <tr> <td>QA</td> <td>@ Kevin Campbell</td> </tr> </table> <div style="display: flex; justify-content: space-between;"> <div style="width: 60%;"> <p>Background and strategic fit</p> <p>We all know mobile is on the rise. A recent survey to customers showed that 85% of users use their mobile on a daily basis. Most of our customers also use competitor apps, so this is something we need to have.</p> <p>Customer research</p> <ul style="list-style-type: none"> Customer interview - Netflix Customer interview - Homeaway Customer interview - Bitbucket </div> <div style="width: 35%;"> <p>Requirements</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>#</th> <th>User story title</th> <th>User story description</th> <th>Priority</th> <th>Notes</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Facebook Integration MDT-13 TO DO</td> <td>A user wants to sign up via Facebook</td> <td>Must Have</td> <td> <ul style="list-style-type: none"> We will need to talk to Cassie Owens. There has also been some research done on this (see Facebook integration prototype) </td> </tr> <tr> <td>2</td> <td>Activity Stream MDT-14 TO DO</td> <td>A user wants to view the latest updates via the mobile dashboard so that they can get a better understanding of what is in place</td> <td>Must Have</td> <td></td> </tr> <tr> <td>3</td> <td>Post Updates MDT-15 TO DO</td> <td>A user wants to be able to post status updates on the go</td> <td>Must Have</td> <td> The key things we will need to support: <ul style="list-style-type: none"> Text status updates Mentions Support for images Smart embedding for YouTube vids </td> </tr> <tr> <td>4</td> <td>API MDT-16 TO DO</td> <td>A developer wants to integrate with the mobile app so that they can embed the activity stream on their website</td> <td>Should Have</td> <td> <ul style="list-style-type: none"> We should chat to Team Dyno as they did something similar. </td> </tr> </tbody> </table> </div> </div> </div>	Target release	1.0	Epic	MDT-18 - Mobile optimized web app TO DO	Document status	DRAFT	Document owner	@ Mitch Davis	Designer	@ Cassie Owens	Developers	@ Harvey Jennings	QA	@ Kevin Campbell	#	User story title	User story description	Priority	Notes	1	Facebook Integration MDT-13 TO DO	A user wants to sign up via Facebook	Must Have	<ul style="list-style-type: none"> We will need to talk to Cassie Owens. There has also been some research done on this (see Facebook integration prototype) 	2	Activity Stream MDT-14 TO DO	A user wants to view the latest updates via the mobile dashboard so that they can get a better understanding of what is in place	Must Have		3	Post Updates MDT-15 TO DO	A user wants to be able to post status updates on the go	Must Have	The key things we will need to support: <ul style="list-style-type: none"> Text status updates Mentions Support for images Smart embedding for YouTube vids 	4	API MDT-16 TO DO	A developer wants to integrate with the mobile app so that they can embed the activity stream on their website	Should Have	<ul style="list-style-type: none"> We should chat to Team Dyno as they did something similar.
Target release	1.0																																							
Epic	MDT-18 - Mobile optimized web app TO DO																																							
Document status	DRAFT																																							
Document owner	@ Mitch Davis																																							
Designer	@ Cassie Owens																																							
Developers	@ Harvey Jennings																																							
QA	@ Kevin Campbell																																							
#	User story title	User story description	Priority	Notes																																				
1	Facebook Integration MDT-13 TO DO	A user wants to sign up via Facebook	Must Have	<ul style="list-style-type: none"> We will need to talk to Cassie Owens. There has also been some research done on this (see Facebook integration prototype) 																																				
2	Activity Stream MDT-14 TO DO	A user wants to view the latest updates via the mobile dashboard so that they can get a better understanding of what is in place	Must Have																																					
3	Post Updates MDT-15 TO DO	A user wants to be able to post status updates on the go	Must Have	The key things we will need to support: <ul style="list-style-type: none"> Text status updates Mentions Support for images Smart embedding for YouTube vids 																																				
4	API MDT-16 TO DO	A developer wants to integrate with the mobile app so that they can embed the activity stream on their website	Should Have	<ul style="list-style-type: none"> We should chat to Team Dyno as they did something similar. 																																				

	<p>User interaction and design</p> <table border="1"> <thead> <tr> <th>Description</th> <th>Login screen</th> <th>Activity stream</th> </tr> </thead> <tbody> <tr> <td>Mockup</td> <td></td> <td></td> </tr> </tbody> </table> <p>Questions</p> <p>Below is a list of questions to be addressed as a result of this requirements document:</p> <table border="1"> <thead> <tr> <th>Question</th> <th>Outcome</th> </tr> </thead> <tbody> <tr> <td>What about Google Apps</td> <td> <ul style="list-style-type: none"> We think this is important, but not for version one. We can look at this at a later stage. 💡 It might be worth someone looking into a shared notification library to do this. </td> </tr> <tr> <td>Are we supporting Blackberry?</td> <td> <ul style="list-style-type: none"> Again, not for initial version - but we haven't had much demand for this. </td> </tr> <tr> <td>Should we have an offline mode?</td> <td> <ul style="list-style-type: none"> We've talked about the pros and cons. In brief: <ul style="list-style-type: none"> ➕ Seamless experience for customers, they won't notice if there is a connection issue ➕ Most of our competitors don't have this ➖ Could be expensive to build ❓ Should we spike this at a later sprint? </td> </tr> </tbody> </table> <p>Not Doing</p> <ul style="list-style-type: none"> Google Apps Authentication - out of scope, see above for details Blackberry support - we won't look at doing this, if demand picks up we can look at it. Native app. We are starting with a mobile web view first and get back to a native app depending on feedback that we get. <p>👍 Like Be the first to like this requirements ✎</p>	Description	Login screen	Activity stream	Mockup			Question	Outcome	What about Google Apps	<ul style="list-style-type: none"> We think this is important, but not for version one. We can look at this at a later stage. 💡 It might be worth someone looking into a shared notification library to do this. 	Are we supporting Blackberry?	<ul style="list-style-type: none"> Again, not for initial version - but we haven't had much demand for this. 	Should we have an offline mode?	<ul style="list-style-type: none"> We've talked about the pros and cons. In brief: <ul style="list-style-type: none"> ➕ Seamless experience for customers, they won't notice if there is a connection issue ➕ Most of our competitors don't have this ➖ Could be expensive to build ❓ Should we spike this at a later sprint?
Description	Login screen	Activity stream													
Mockup															
Question	Outcome														
What about Google Apps	<ul style="list-style-type: none"> We think this is important, but not for version one. We can look at this at a later stage. 💡 It might be worth someone looking into a shared notification library to do this. 														
Are we supporting Blackberry?	<ul style="list-style-type: none"> Again, not for initial version - but we haven't had much demand for this. 														
Should we have an offline mode?	<ul style="list-style-type: none"> We've talked about the pros and cons. In brief: <ul style="list-style-type: none"> ➕ Seamless experience for customers, they won't notice if there is a connection issue ➕ Most of our competitors don't have this ➖ Could be expensive to build ❓ Should we spike this at a later sprint? 														
<p>Elementos de la problemática a los que se relaciona o solventa:</p>	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia de personas externas. - Dependencia entre requisitos. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos. 														
<p>Fuente de información:</p>	<p>Atlassian (2021), <i>Agile Coach</i></p>														

Identificador:	L-012
Nombre de la práctica:	Hoja de ruta del producto
Descripción de la Práctica:	<p>La hoja de ruta de un producto es una fuente de información compartida que describe la visión, dirección y progreso de un producto en el transcurso del tiempo.</p> <ul style="list-style-type: none"> - Incluir solamente los detalles necesarios para el público - Enfocar la hoja de ruta en las tácticas a corto plazo y en su vinculación a los objetivos a largo plazo por igual - Revisar las hojas de ruta con regularidad y realizar ajustes cuando cambian los planes - Asegurarse de que los interesados tienen acceso a la hoja de ruta. - Conectarse con los interesados para garantizar la coordinación
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Dependencia de personas externas. - Dependencia entre requisitos. - Cambio de prioridad durante el sprint. - Cambios no gestionados en los requisitos.
Fuente de información:	Atlassian (2021), <i>Agile Coach</i>

Identificador:	L-013
Nombre de la práctica:	Redacción de historias de usuario
Descripción de la Práctica:	<p>Las historias de usuario suelen expresarse con la siguiente estructura:</p> <p style="text-align: center;">“Como [perfil], [quiero] [para].”</p> <ul style="list-style-type: none"> - “Como [perfil]”: Se define el perfil de la persona que será el usuario del sistema. Se debe comprender cómo trabaja esa persona, cómo piensa y cómo se siente. - “Quiere”: Se describe lo que se intenta lograr realmente. Esta descripción representa el objetivo del usuario no la funcionalidad de la solución. - “Para”: Se describe el beneficio que desea alcanzarse o el problema que desea resolverse <p>Esta estructura resulta de ayuda para establecer una definición de "hecho". Cuando ese perfil puede alcanzar su valor deseado, la historia está completa.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Ambigüedad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Atlassian (2021), <i>Agile Coach</i>

Identificador:	L-014
Nombre de la práctica:	Descomposición del trabajo en Jira
Descripción de la Práctica:	<p>Al planificar un proyecto grande, se requiere descomponer el trabajo por realizar desde los objetivos más ambiciosos hasta los detalles más minuciosos, manteniendo un balance entre estructuración y flexibilidad. Para esto puede utilizarse el siguiente desglose del trabajo.</p> <ul style="list-style-type: none"> - Las iniciativas son conjuntos de épicas que conducen hacia un objetivo común. - Las épicas son grandes cantidades de trabajo que se pueden desglosar en un número de tareas más pequeñas (llamadas "historias"). - Las historias, también llamadas "historias de usuario", son breves requisitos o solicitudes escritas desde el punto de vista del usuario final. 
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos. - Dependencia entre requisitos.
Fuente de información:	Atlassian (2021), <i>Agile Coach</i> .

Identificador:	L-015
Nombre de la práctica:	Estimación de historias de usuario
Descripción de la Práctica:	<p>La estimación consiste en determinar el esfuerzo requerido para entregar el trabajo solicitado. Los puntos de historia son unidades de medida que permiten expresar una estimación del esfuerzo total que deberá hacer el equipo para implementar íntegramente un elemento del <i>backlog</i> del producto. Los equipos asignan puntos de historia en función de la complejidad y del volumen del trabajo, así como del riesgo o de la incertidumbre.</p> <p>Los miembros del equipo toman un elemento del <i>backlog</i>, hablan sobre él brevemente y cada uno fórmula mentalmente una estimación. A continuación, todos levantan una tarjeta con el número que refleje su estimación. Si las estimaciones coinciden se continúa con el siguiente elemento del <i>backlog</i>. Si no, se dedica tiempo para entender el motivo de las diferencias.</p>
Elementos de la problemática a los que se relaciona o solventa:	<ul style="list-style-type: none"> - Complejidad de los requisitos.
Fuente de información:	Atlassian (2021), <i>Agile Coach</i> .

8.13 Apéndice M. Plantilla del plan de gestión de requisitos

La plantilla incluye el formato a utilizar y un ejemplo del cómo debe completarse cada actividad.

Plan de gestión de requisitos

Datos del Proyecto

Nombre del Proyecto	Product Owner Asignado	Equipo Scrum asignado
Proyecto Innovación	Juan Pérez	Equipo Alpha

Actividades de gestión de requisitos

¿Qué?	¿Cuándo?	¿Dónde?	¿Quién facilita?	¿Quiénes participan?
Taller de definición de roles de usuario	Todos los martes a las 2pm	Sala 3. Zoom ID: 859624123	Product Owner	Equipo Scrum, Representante del cliente, Especialista invitado.

8.14 Apéndice N. Plantilla del documento de requisitos del producto (PRD)

Confluence Espacios Personas Crear ... Buscar

Panel / Página de inicio de Mayid Nasralah Calderon

Documento de requisitos del producto

Creado por Mayid Nasralah Calderon, modificado por última vez en ago 29, 2021

Datos del Proyecto

Nombre del Proyecto	Product Owner Asignado	Equipo Scrum asignado

Enunciado de la Necesidad

Descripción concreta de la necesidad que se desea satisfacer desde el punto de vista del negocio.

Matriz de interesados y expectativas

Nombre del Interesado	Rol del interesado	Datos de contacto	Expectativas
			<input type="checkbox"/> Expectativa 1 <input type="checkbox"/> Expectativa 2
			<input type="checkbox"/> Expectativa 1 <input type="checkbox"/> Expectativa 2
			<input type="checkbox"/> Expectativa 1 <input type="checkbox"/> Expectativa 2

Matriz de trazabilidad de requisitos

Requisito de 1er nivel	Requisito de 2do nivel	Requisito de 3er nivel	Material documental o gráfico	Estimación	Priorización
Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Se adjunta el documento o material gráfico.	<input type="checkbox"/> XS <input type="checkbox"/> S <input type="checkbox"/> M <input type="checkbox"/> L <input type="checkbox"/> XL	<input type="checkbox"/> Imprescindible <input type="checkbox"/> Importante <input type="checkbox"/> Deseable <input type="checkbox"/> Prescindible
		Requisito de 3er nivel Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Se adjunta el documento o material gráfico.	<input type="checkbox"/> XS <input type="checkbox"/> S <input type="checkbox"/> M <input type="checkbox"/> L <input type="checkbox"/> XL	<input type="checkbox"/> Imprescindible <input type="checkbox"/> Importante <input type="checkbox"/> Deseable <input type="checkbox"/> Prescindible
	Requisito de 2do nivel Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Requisito de 3er nivel Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Se adjunta el documento o material gráfico.	<input type="checkbox"/> XS <input type="checkbox"/> S <input type="checkbox"/> M <input type="checkbox"/> L <input type="checkbox"/> XL	<input type="checkbox"/> Imprescindible <input type="checkbox"/> Importante <input type="checkbox"/> Deseable <input type="checkbox"/> Prescindible
		Requisito de 3er nivel Historia de usuario: "Como" + rol de usuario + "requiero" + expectativa + "para" + razón. Criterios de aceptación: <input type="checkbox"/> Criterio de aceptación 1 <input type="checkbox"/> Criterio de aceptación 2	Se adjunta el documento o material gráfico.	<input type="checkbox"/> XS <input type="checkbox"/> S <input type="checkbox"/> M <input type="checkbox"/> L <input type="checkbox"/> XL	<input type="checkbox"/> Imprescindible <input type="checkbox"/> Importante <input type="checkbox"/> Deseable <input type="checkbox"/> Prescindible

Pregunta generadoras

Preguntas que aclaren el alcance de la solución y complementen la matriz de trazabilidad de requisitos.

Registro de cambios en los requisitos

Historial de cambios solicitados a los requisitos y su resolución.

Fecha de solicitud	Solicitante	Descripción del cambio	Aprobación o rechazo	Fecha de aprobación o rechazo	Comentarios

 Me gusta Sé el primero al que le gusta esto

Sin etiquetas 



Escribe un comentario...

8.15 Apéndice O. Diccionario de términos para la gestión de requisitos

A.

Adaptativo: Proceso o requisito que se adapta según la necesidad del producto o solución.

Ambiente: Son los factores de aspectos políticos, de mercado, culturales y organizacionales que influyen en los requisitos.

Análisis de requisitos: Consiste en la examinación, descomposición y síntesis de la información de los requisitos obtenida, en busca de alcanzar un conjunto accionable de requisitos que satisfaga las metas y objetivos planteados.

Atributos: Son características específicas que capturan información clave del requisito.

C.

Chief Product Owner: Líder del *MetaScrum*. Persona responsable de priorizar las iniciativas de nuevos productos y las iniciativas de mejora de productos existentes.

Ciclo de vida predictivo: Ciclo de vida en el cual la obtención y análisis de requisitos se realiza previamente al inicio del diseño y desarrollo de la solución.

Ciclo de vida adaptativo: Ciclo de vida en el cual la obtención y análisis de requisitos se realiza en el transcurso del proyecto, lo que provee mayor oportunidad de identificar las necesidades de negocio y comprender la solución requerida. Este tipo de ciclo de vida implica un proceso de requisitos iterativo.

Cierre del proyecto o fase: Se realiza una vez el producto, servicio o resultado ha sido enviado a un estado operacional y las actividades de evaluación de la solución han sido ejecutadas satisfactoriamente entregando el valor esperado al negocio.

Cliente: Persona o empresa que provee los objetivos, necesidades o problemas al proceso de desarrollo de requisitos de sistemas.

Comunidad técnica: Personas con alto nivel técnico que clarifican y confirman los requisitos desde su criterio técnico experto.

Confluence: Es una herramienta de *software* complementaria de Jira, que facilita el dar seguimiento a los requisitos del proyecto.

D.

Daily Scrum: Sesión diaria, no mayor a 15 minutos que consiste en revisar el progreso hacia el objetivo del *sprint* y adaptar el *sprint backlog*.

Datos de negocio: Documentos que permiten identificar los factores que rodean la problemática u oportunidad de negocio.

Descomposición: Consiste en descomponer los requisitos en otros más discretos de forma que sea disminuida la ambigüedad.

Developers: Son los miembros del equipo *Scrum* responsables de crear el incremento utilizable en cada *sprint*.

Documentos de proyectos: Documentos que sirven como insumo para la planificación de la gestión de requisitos con base en la experiencia adquirida de proyectos similares.

Documento de requisitos del producto (PRD): Consiste en un documento de una sola página que unifica y define los requisitos del producto concreto. Los requisitos deben permitir una clara comprensión de las funcionalidades del producto y su comportamiento durante todo el desarrollo de la solución.

E.

Enunciado de la necesidad: Consiste en una descripción concreta de la necesidad que se desea satisfacer desde el punto de vista del negocio.

Especificación del requisito: Contiene todos los requisitos de software con el nivel de detalle suficiente para permitir un diseño que los satisfaga y unas pruebas que los verifiquen.

Evaluación de la necesidad: Identificación y definición de un problema u oportunidad de negocio. Con el fin de obtener una definición de alto nivel de la necesidad que permita determinar una o varias alternativas de solución.

Evaluación de la solución: Se compone de las actividades desarrolladas para validar una solución que será o ha sido implementada. Determina que tan bien la solución satisface las necesidades de negocio.

F.

Formulación de requisitos: Consiste en asegurar que cada requisito es necesario, corto, comprobable y que representa una necesidad.

Funciones del producto: Provee un resumen de las funciones principales del software, usualmente tomadas de la sección de alto nivel de la especificación.

G.

Gestión de cambio: Proceso que asegura que los cambios son gestionados y los requisitos son actualizados una vez que el cambio es aprobado.

Gestión de la comunicación: Incluye las actividades requeridas para asegurar la apropiada planificación, recolección, creación, distribución, almacenamiento, gestión, control y monitoreo de la información del proyecto.

Gestión de interesados: Consiste en las actividades para identificar a todas aquellas personas y organizaciones impactadas por el proyecto, analizar sus características y desarrollar estrategias que fomenten su involucramiento.

Gestión de requisitos: Consiste en todas aquellas actividades para desarrollar y gestionar los requisitos de un proyecto, que permiten establecer una línea base, trazabilidad y control de cambios de los requisitos.

I.

Identificación de la necesidad: Consiste en identificar, evaluar la importancia y limitar las necesidades de negocio.

Identificación de requisitos: Consiste en extraer de los clientes, la comunidad técnica y el ambiente un conjunto de requisitos.

Interesados: Todo aquel individuo, grupo u organización que tiene una relación directa o indirecta con la solución, durante y después de su desarrollo.

Iterativo: Consiste en repetir las actividades del proyecto en ciclos.

J.

Jira: es un software propiedad de la compañía Atlassian, utilizado principalmente por equipos ágiles para la gestión del proceso de desarrollo de *software*, pero sus funcionalidades le permiten extender su alcance a diversas metodologías de gestión de proyectos.

M.

Matriz de interesados y expectativas: Consiste en un cuadro que relaciona a los interesados con sus expectativas. Además, incluye información de contacto de cada uno de los interesados.

Matriz de trazabilidad: Captura los atributos de los requisitos durante la etapa del análisis.

MetaScrum: Grupo de personas compuesto por los *Product Owners* de cada equipo *Scrum* y el *Chief Product Owner*; y tiene como objetivo gestionar la cartera de proyectos.

Modelos: Son representaciones gráficas o textuales que permiten identificar información extraña o ausencia de información.

Monitoreo y control de requisitos: Consiste en rastrear los requisitos de manera que se asegure que el proyecto satisface el alcance del producto, servicio o resultado y que los cambios en los requisitos son incluidos en el alcance cuando son aprobados.

O.

Obtención de los requisitos: Consiste en el proceso de extracción e identificación de la información necesaria para desarrollar los requisitos de la solución.

Organización de requisitos: Consiste en crear una estructura para el conjunto de requisitos estableciendo las relaciones entre ellos.

P.

Plan de gestión de requisitos: Documento que define como evolucionan los requisitos reales y como se abordarán las actividades de gestión de requisitos.

Planificación de la gestión de requisitos: Consiste en brindar guía y dirección del cómo serán desarrollados y gestionados los requisitos durante el proyecto.

Presentación de requisitos: Consiste en determinar las mejores maneras de comunicar los requisitos a cada interesado que requiera entender, revisar, aceptar o utilizar las especificaciones.

Priorización: Consiste en ordenar los Requisitos según su importancia en busca de facilitar el análisis del valor relativo de un requisito con respecto a otro.

Proceso de gestión de requisito: Conjunto de actividades estandarizadas y estructuradas para desarrollar y gestionar los requisitos de un proyecto.

Product backlog: Es una lista emergente y ordenada de lo que se necesita para mejorar el producto.

Es la única fuente del trabajo realizado por el equipo *Scrum*.

Product Owner: Miembro del equipo *Scrum* responsable de maximizar el valor del producto resultante del trabajo del equipo *Scrum*, de desarrollar y comunicar explícitamente el objetivo del producto

Prorrateo de requisitos: identifica requisitos que deberán esperar hasta que estén disponible nuevas versiones del sistema.

Prototipo: Consiste en un modelo que permite observar aspectos no contemplados del comportamiento del *software* y así generar menos cambios en la especificación.

R.

Refinamiento: Revisión continua de los elementos del *product backlog* dentro de cada *sprint* con el fin de tenerlos listos para futuros *sprints*.

Requisito: Es una condición o capacidad que se requiere esté presente en el producto, servicio o resultado para satisfacer un contrato u otra especificación formal.

Requisito completo: Que incluye todos los requisitos significativos relacionados a la funcionalidad, desempeño, restricciones de diseño, atributos e interfaces externas. También debe incluir la definición de las respuestas del software a toda clase de entrada en cualquier situación.

Requisito correcto: Que está presente en el *software*, para garantizar que se satisface la necesidad.

Requisito consistente: Que no presentan conflictos entre sí, están priorizados por importancia, necesidad o estabilidad.

Requisito inequívoco: Que no presenta ambigüedad y son interpretados siempre de la misma manera.

Requisito modificable: Que permite realizar cambios de forma sencilla, completa y consistente manteniendo su estructura y estilo.

Requisito rastreable: Que su origen es claro y facilita referenciarlo en desarrollos futuros o en la documentación.

Requisito verificable: Que es posible verificar que el producto de software cumple con lo establecido en el requisito, evitando términos subjetivos para describir dicha verificación.

Restricción: Es un requisito impuesto a la solución por una circunstancia, fuerza o coacción, el cual impone límites que deben ser respetados.

S.

Scrum: Marco de trabajo ágil diseñado para resolver problemas complejos mediante soluciones adaptativas, basado en el empirismo y en el pensamiento *Lean*.

Scrum Master: miembro del equipo *Scrum* responsable de implementar *Scrum* como se define en la Guía de *Scrum* y de guiar a los miembros del equipo para generar valor.

Scrum of Scrums: Compuesto por los *Scrum Master* de cada equipo *Scrum* y el *Chief Scrum Master*; y tiene como objetivo mantener y mejorar los procesos de *Scrum*.

Software: Conjunto de programas y rutinas que permiten al sistema realizar determinadas tareas.

Sprint: Evento *Scrum* que contiene al resto de eventos, tiene una duración fija menor a un mes e inicia inmediatamente después de la conclusión del *sprint* anterior.

Sprint planning: Evento *Scrum* en el cual se establece el plan de trabajo que se realizará durante el *sprint*. El equipo *Scrum* crea este plan mediante trabajo colaborativo.

Sprint review: Evento *Scrum* en el cual se inspecciona el resultado del *sprint* y se determinan futuras adaptaciones del producto.

Sprint retrospective: Evento *Scrum* en el cual planifican formas de aumentar la calidad y la efectividad de los procesos *Scrum*.

Supuestos: Lista aquellos factores externos que se asume afectarán los requisitos estipulados.

T.

Taller de definición de roles de usuario: Consiste en una sesión de trabajo conjunto entre el equipo *Scrum*, los interesados clave y demás invitados para identificar con certeza cada uno de los roles de usuario que interactúan con el producto.

Taller de identificación de requisitos: Consiste en una sesión de trabajo entre el equipo *Scrum*, los interesados clave y demás invitados para identificar los requisitos de alto nivel desde la perspectiva de cada uno de los roles de usuario.

Taller de formulación de historias de usuario: Consiste en una sesión de trabajo entre el equipo *Scrum*, los interesados clave y demás invitados para formular los requisitos de alto nivel mediante la técnica de historia de usuario desde la perspectiva de cada uno de los roles de usuario.

Taller de descomposición: Consiste en una sesión de trabajo entre el equipo *Scrum* y demás invitados para descomponer los requisitos de alto nivel en requisitos puntuales, únicos e independientes, con el fin de disminuir su complejidad y ambigüedad al definir con mayor exactitud las expectativas y los criterios de aceptación.

Taller de estimación: Consiste en una sesión de trabajo exclusiva del equipo *Scrum* para definir el esfuerzo en puntos que será necesario para desarrollar cada requisito.

Taller de priorización: Consiste en una sesión de trabajo entre el equipo *Scrum* y demás invitados para categorizar los requisitos según su aporte a la satisfacción de las expectativas de los interesados, con el fin de generar el mayor valor de negocio en la menor cantidad de iteraciones posible.

Taller de verificación y validación: Consiste en una sesión de trabajo exclusiva del equipo *Scrum* para obtener retroalimentación del *Product Owner* con respecto al producto desarrollado, con el

fin de identificar si es idóneo para su puesta en producción o se requiere de modificación de sus requisitos para futuras iteraciones.

V.

Validación: Consiste en asegurar que los requisitos reflejen las intenciones de los interesados y que permitan satisfacer sus expectativas.

Verificación: Consiste en revisar los requisitos para asegurar que el producto ha sido correctamente desarrollado y está libre de errores y omisiones.