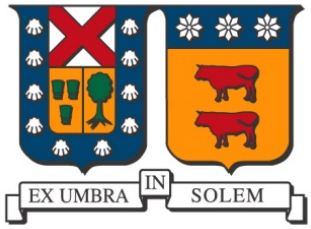
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
VALPARAÍSO – CHILE



“PROPUESTA DE ADECUACIÓN DEL FLUJO DE PROCESOS ÁGIL DE LA FERIA DE SOFTWARE A LA REALIDAD ACTUAL DE SUS PROYECTOS”

PAULA ALEJANDRA CASTILLO OJEDA

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERA CIVIL EN INFORMÁTICA

Profesor Guía: Pedro Godoy Barrera
Profesora Correferente: Luz Chourio

Octubre - 2026

CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado

Título del trabajo: Propuesta de adecuación del flujo de procesos ágil de la feria de software a la realidad act

Nombre del candidato(a): Paula Alejandra Castillo Ojeda

Carrera / Grado: Ingeniería Civil Informática

Campus: __Casa Central _____ **Departamento:** _____Departamento de Informática_____

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, __Pedro Francisco Godoy Barrera_____, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por **(marcar una opción)**:

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

_____

_____

_____

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 21-Oct-2026 **Firma:** 

Estudiante o Candidato(a):

Fecha: _____20-10-2026_____ **Firma:** 

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

DEDICATORIA

Para mi mamá, que nunca me voy a cansar de decirle lo orgullosa que estoy de ella y de todo lo que ha hecho por mí. Gracias por enseñarme a ser fuerte, a no rendirme y a luchar por mis sueños. Gracias por estar siempre ahí para apoyarme y darme la confianza para seguir adelante.

AGRADECIMIENTOS

Primero, quiero agradecer a mi profesor guía, Pedro Godoy, quien me ha acompañado durante todo este camino, incluso desde antes de comenzar la memoria. Gracias por ayudarme a reforzar la constancia, la perseverancia y la organización, en especial durante mi paso por el centro de estudiantes, y por su paciencia, su apoyo y la confianza que siempre ha depositado en mí.

También quiero agradecer a Wladimir Ormazabal, que siempre ha creído en mí y con quien he contado incontables veces. No habría llegado hasta aquí sin su ayuda y su guía. Gracias por enseñarme a ser una mejor profesional.

A la profesora Luz Chourio, por aceptar ser correferente de este trabajo y por sus comentarios, que ayudaron a mejorarlo.

A mi familia. A mi mamá, Orietta Ojeda, que siempre estuvo ahí para recordarme lo orgullosa que estaba de mí, y a Luis Oteiza, que siempre ha creído en mí. A mi hermano, Ignacio Castillo, que sin saberlo me preparó toda la vida para el estrés de este momento, y a mi papá, Sergio Castillo, por estar presente durante este proceso. A mi abuela, María Angélica Cerda, y a mi tata, José Ojeda, que nunca me han dejado sola y siempre han tenido una mano para ofrecerme. Y a mi prima, Renata Ojeda, que ha estado ahí cada vez que la he necesitado.

A mis queridos amigos del Tryhard, con quienes recorrimos juntos todos estos años y que fueron una parte fundamental para llegar hasta aquí. Los llevaré en mi corazón toda la vida. Una mención especial para Martina Tejo, mi confidente y amiga durante todo este tiempo, con quien he reído, llorado y amanecido gran parte de estos años.

A Javier Espíndola, que aunque llegó al final del proceso no es menos importante. Gracias por acompañarme en la etapa más difícil de la carrera y por tomar mi mano cada vez que necesité un respiro.

A mis amigas de la vida, Vale, Cata, Nachi y Coni, que han estado para apoyarme en los momentos difíciles y para celebrar conmigo los momentos felices.

A Envision, por creer en mí y permitirme ser parte de su hermosa organización. Gracias por valorarme como profesional.

Finalmente, quiero agradecerme a mí misma, porque solo yo sé lo que me costó llegar hasta acá, y todo lo vivido valió la pena para llegar a este punto. Gracias, Pali.

RESUMEN

Resumen—Este trabajo propone una adecuación del flujo de procesos ágil de la Feria de Software de la UTFSM (asignaturas GPI y TDPI) a la realidad actual de sus proyectos. El marco vigente, definido en 2019 y basado en Scrum, no contempla la diversificación de los proyectos hacia soluciones centradas en un modelo de Machine Learning, el uso de herramientas de inteligencia artificial en el desarrollo ni las mayores exigencias de innovación y emprendimiento del nuevo plan de estudios. A partir del análisis de ese marco, de la literatura, de la experiencia de la autora como estudiante y ayudante, y de un levantamiento de los proyectos presentados entre 2018 y 2025, se documentó una propuesta que conserva las ceremonias, los roles y el calendario del marco vigente. La propuesta clasifica los proyectos según su naturaleza e integra CRISP-DM en los proyectos que entrenan un modelo propio, reemplaza el Plan de Proyecto por un Documento de Diseño que se actualiza en cada sprint y formaliza un proceso de emprendimiento. Se validó con un especialista del 3IE y con una encuesta a 63 estudiantes, ayudantes y profesores, que validó seis de las siete propuestas, tres de ellas con observaciones de factibilidad, y en la que el 90 % de los participantes la aplicaría o recomendaría, completa o en parte.

Palabras Clave—Feria de Software; adecuación de procesos; metodologías ágiles; proyectos de Machine Learning; emprendimiento.

ABSTRACT

***Abstract*—**This work proposes an adaptation of the agile process flow of the UTFSM Software Fair (Feria de Software), which spans the courses GPI and TDPI, to the current reality of its projects. The existing framework, defined in 2019 and based on Scrum, does not account for the shift of projects toward solutions built around a Machine Learning model, the use of artificial intelligence tools in development, or the greater innovation and entrepreneurship requirements of the new curriculum. Based on an analysis of that framework, the literature, the author's experience as a student and teaching assistant, and a review of the projects presented between 2018 and 2025, a proposal was documented that keeps the ceremonies, roles and schedule of the existing framework. The proposal classifies projects by their nature and integrates CRISP-DM into projects that train their own model, replaces the Project Plan with a Design Document that is updated every sprint, and formalizes an entrepreneurship process. It was validated with a specialist from the 3IE and through a survey of 63 students, teaching assistants and professors, which validated six of the seven proposals, three of them with feasibility observations, and in which 90% of participants would apply or recommend it, fully or in part.

Keywords—Software Fair; process adaptation; agile methodologies; Machine Learning projects; entrepreneurship.

GLOSARIO

3IE: Instituto Internacional para la Innovación Empresarial de la UTFSM.
Ayudante: Alumno que ya cursó la asignatura y guía a los equipos con su experiencia previa.
BPMN: *Business Process Model and Notation*, notación gráfica estándar para modelar procesos mediante actividades, compuertas y carriles.
Business Model Canvas: Lienzo de nueve bloques que describe cómo un negocio crea, entrega y captura valor.
CRISP-DM: *Cross-Industry Standard Process for Data Mining*, ciclo de vida de seis fases para proyectos de minería de datos y Machine Learning.
CTR: *Click-Through Rate*, porcentaje de personas que hacen clic en un contenido respecto del total que lo vio.
Demoday: Instancia del evento de Feria de Software en que cada equipo presenta su proyecto como startup mediante un pitch ante jueces externos o internos.
Design Thinking: Metodología de innovación centrada en las personas, que avanza por etapas de empatizar, definir, idear, prototipar y evaluar.
DI: Departamento de Informática.
FESW: Feria de Software.
GPI: Gestión de Proyectos de Informática.
HdU: Historia de Usuario.
IA: Inteligencia Artificial.
KPI: *Key Performance Indicator*, indicador clave que mide el avance hacia un objetivo.
Lean Canvas: Adaptación del Business Model Canvas para etapas tempranas, centrada en el problema, el cliente, la solución y la propuesta de valor.
Manual de marca (brand book): Documento que reúne la identidad de una marca y las reglas para usarla de forma consistente.
ML: Machine Learning.
MLOps: Prácticas para desplegar, monitorear y mantener modelos de Machine Learning en producción.
NABC: Necesidad, Aproximación, Beneficios y Competencia. Método para validar una propuesta de valor.
ODS: Objetivos de Desarrollo Sostenible.
PMBOK: Project Management Body of Knowledge.
PMV: Producto Mínimo Viable, versión de un producto con las funcionalidades mínimas para ser usado por clientes reales. En GPI se usa coloquialmente para el prototipo inicial.
SCT: Sistema de Créditos Transferibles.
Spike: Actividad de investigación acotada en el tiempo que busca reducir la incertidumbre técnica o funcional antes de comprometer el desarrollo.
TDPI: Taller de Desarrollo de Proyectos de Informática.
UTFSM: Universidad Técnica Federico Santa María.

ÍNDICE DE CONTENIDOS

RESUMEN ..... **iii**

ABSTRACT ..... **iii**

GLOSARIO ..... **v**

ÍNDICE DE FIGURAS ..... **ix**

ÍNDICE DE TABLAS ..... **ix**

INTRODUCCIÓN **1**

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA **3**

1.1 CONTEXTO 3

1.1.1 FERIA DE SOFTWARE 4

1.2 SITUACIÓN ACTUAL 5

1.3 OBJETIVOS 7

1.3.1 OBJETIVO GENERAL 7

1.3.2 OBJETIVOS ESPECÍFICOS 7

1.3.3 ALCANCE 8

1.4 METODOLOGÍA DEL TRABAJO 8

CAPÍTULO 2: MARCO CONCEPTUAL **10**

2.1 GESTIÓN DE PROYECTOS Y METODOLOGÍAS DE DESARROLLO DE SOFTWARE 10

2.1.1 GESTIÓN DE PROYECTOS 10

2.1.2 METODOLOGÍAS DE DESARROLLO: TRADICIONALES VS. ÁGILES 11

2.1.3 SCRUM 11

2.1.4 ESTIMACIÓN Y SPIKES EN SCRUM 13

2.1.5 METODOLOGÍAS Y HERRAMIENTAS ÁGILES EN LA INDUSTRIA 14

2.2 CURSOS CAPSTONE Y APRENDIZAJE BASADO EN PROYECTOS 15

2.2.1 METODOLOGÍAS ÁGILES Y RESTRICCIONES DEL ENTORNO ACADÉMICO 16

2.2.2 ETAPAS Y ROLES DEL MARCO VIGENTE 17

2.3 INNOVACIÓN Y EMPRENDIMIENTO EN PROYECTOS DE SOFTWARE 19

2.3.1 EL ENFOQUE DE INNOVACIÓN EN LA FERIA DE SOFTWARE 2025 21

2.3.2 INNOVACIÓN SIN EMPRENDIMIENTO 22

2.3.3 HERRAMIENTAS PARA MODELAR Y COMUNICAR UN EMPRENDIMIENTO 23

2.4 IA COMO HERRAMIENTA DE DESARROLLO 25

2.4.1 IMPACTO DE LA IA EN EL CICLO DE DESARROLLO 25

2.5 DIVERSIFICACIÓN DE PROYECTOS 26

2.5.1 MODELOS DE PROCESO PARA PROYECTOS DE DATOS 27

2.5.2 CICLO DE VIDA DE UN PROYECTO DE MACHINE LEARNING 29

2.6 DATOS PERSONALES EN PROYECTOS ACADÉMICOS 30

2.7 ADAPTACIÓN DE MARCOS ÁGILES 31

2.8 TRABAJOS RELACIONADOS 32
2.9 SÍNTESIS DE LA PROBLEMÁTICA 33
2.9.1 LO QUE PLANTEA LA LITERATURA 33
2.9.2 LA EXPERIENCIA COMO ESTUDIANTE Y COMO AYUDANTE 34
2.9.3 EVOLUCIÓN DE LOS PROYECTOS DE LA FERIA DE SOFTWARE 35
2.9.4 DESAJUSTES IDENTIFICADOS 36

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN 38
3.1 ENFOQUE DE LA PROPUESTA 38
3.2 ESTRUCTURA PRIMER SEMESTRE 40
3.2.1 PROPUESTA DE INNOVACIÓN Y DEFINICIÓN DEL PROBLEMA 41
3.2.2 DEFINICIÓN DEL ALCANCE Y RIESGOS 44
3.2.3 PLANIFICACIÓN Y DISEÑO 52
3.3 ESTRUCTURA SEGUNDO SEMESTRE 58
3.3.1 SPRINT 1: DESARROLLO INICIAL 59
3.3.2 SPRINT 2: DESARROLLO CON RETROALIMENTACIÓN Y PUNTO DE DECISIÓN .. 65
3.3.3 SPRINT 3: CIERRE DEL DESARROLLO 68
3.3.4 SÍNTESIS COMPARATIVA DE AMBOS CAMINOS 70
3.3.5 EVENTO FERIA DE SOFTWARE 72
3.3.6 CIERRE DE PROYECTOS 73
3.3.7 EMPRENDIMIENTO 74
3.4 SÍNTESIS DE LA PROPUESTA 83

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN 86
4.1 DISEÑO DE LA VALIDACIÓN 86
4.2 VALIDACIÓN CON EL INSTITUTO INTERNACIONAL PARA LA INNOVACIÓN EMPRESARIAL 86
4.2.1 OBJETIVO 86
4.2.2 PROCEDIMIENTO 87
4.2.3 RESULTADOS 87
4.2.4 AJUSTES INCORPORADOS 88
4.3 ENCUESTA A PROFESORES, AYUDANTES Y ESTUDIANTES 89
4.3.1 OBJETIVOS DE LA ENCUESTA 89
4.3.2 DISEÑO DEL INSTRUMENTO 89
4.3.3 ANÁLISIS DE LOS DATOS 91
4.3.4 RESULTADOS 91
4.4 SÍNTESIS DE LA VALIDACIÓN 99
4.4.1 LIMITACIONES 101

CAPÍTULO 5: CONCLUSIONES 102
5.1 CUMPLIMIENTO DE LOS OBJETIVOS 102
5.2 APORTES DEL TRABAJO 103
5.3 LIMITACIONES 104

5.4 TRABAJO FUTURO 104
5.5 REFLEXIÓN FINAL 105

ANEXOS **106**

REFERENCIAS BIBLIOGRÁFICAS **114**

ÍNDICE DE FIGURAS

Figura 1 Roles de la propuesta vigente en GPI y TDPI. Fuente: Calvo [1]. 18
Figura 2 Las diez etapas de la propuesta vigente en GPI y TDPI, agrupadas en cuatro categorías. Fuente: Calvo [1]. 19
Figura 3 Modelo del marco de trabajo para la innovación tecnológica propuesto. Fuente: Cayo [2]. 22
Figura 4 Formato de Business Model Canvas utilizado en TDPI. Fuente: Material de la asignatura (Actividad E7). 24
Figura 5 Ciclo de vida CRISP-DM. Fuente: Adaptado y traducido de Wirth y Hipp [3]. . . 29
Figura 6 Procesos del primer semestre. Fuente: Elaboración propia. 40
Figura 7 Proceso de propuesta de innovación y definición del problema. Fuente: Elaboración propia. 41
Figura 8 Proceso de definición del alcance y riesgos. Fuente: Elaboración propia. 45
Figura 9 Proceso de planificación y diseño. Fuente: Elaboración propia. 52
Figura 10 Procesos del segundo semestre. Fuente: Elaboración propia. 58
Figura 11 Proceso de Sprint 1. Fuente: Elaboración propia. 60
Figura 12 Proceso de Sprint 2. Fuente: Elaboración propia. 65
Figura 13 Proceso de Sprint 3. Fuente: Elaboración propia. 69
Figura 14 Proceso de emprendimiento del segundo semestre. Fuente: Elaboración propia. 75
Figura 15 Respuestas a las afirmaciones de diagnóstico. El porcentaje indica las respuestas de acuerdo y muy de acuerdo sin contar «No sé». En las afirmaciones marcadas con asterisco, el desacuerdo es el que confirma el problema. Fuente: Elaboración propia. 93
Figura 16 Respuestas a las afirmaciones de cada propuesta. El porcentaje indica las respuestas de acuerdo y muy de acuerdo sin contar «No sé». Fuente: Elaboración propia. 95
Figura 17 Template del Problema. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM. 106
Figura 18 Template de Segmento de Clientes. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM. 107
Figura 19 Template de Cliente Ideal. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM. 107
Figura 20 Template de la Solución. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM. 108
Figura 21 Template de Propuesta de Valor. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM. 108

Figura 22 Proceso de emprendimiento del segundo semestre, versión dividida. Fuente: Elaboración propia. 109
Figura 23 Sprint 1 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia. 110
Figura 24 Sprint 2 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia. 110
Figura 25 Sprint 3 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia. 111

ÍNDICE DE TABLAS

Tabla 1 Comparación de modelos de proceso para proyectos de datos. Fuente: Elaboración propia. 28
Tabla 2 Proyectos de la Feria de Software según componente de IA/ML, 2018–2025. Fuente: Elaboración propia a partir del sitio oficial de la Feria de Software. 35
Tabla 3 Plantillas del 3IE utilizadas en cada etapa de innovación. Fuente: Elaboración propia a partir de las plantillas del 3IE. 43
Tabla 4 Plantilla propuesta para el Formulario de Spike y la presentación del PMV en proyectos con modelo propio. Fuente: Elaboración propia a partir de las tareas de CRISP-DM. 50
Tabla 5 Plantilla propuesta para el modelo de negocios consolidado. Fuente: Elaboración propia. 54
Tabla 6 Plantilla propuesta para el Documento de Diseño. Fuente: Elaboración propia a partir del Plan de Proyecto del marco vigente. 56
Tabla 7 Distribución propuesta del trabajo en proyectos con modelo propio. Fuente: Elaboración propia a partir de las tareas de CRISP-DM. 57
Tabla 8 Plantilla propuesta para el Formulario de entrega del sprint. Fuente: Elaboración propia. 63
Tabla 9 Plantilla propuesta para el registro de cambios de la Actualización de documentación. Fuente: Elaboración propia. 65
Tabla 10 Síntesis comparativa del contenido de cada etapa según la naturaleza del proyecto. Fuente: Elaboración propia. 71
Tabla 11 Plantilla propuesta para el Documento de Legado. Fuente: Elaboración propia a partir de Calvo [1]. 74
Tabla 12 Plantilla propuesta para el manual de marca. Fuente: Elaboración propia. 77
Tabla 13 Partes del Plan de Marketing y su relación con el modelo de diez etapas. Fuente: Elaboración propia a partir del material de la asignatura. 78
Tabla 14 Plantilla propuesta para la definición de KPI y metas por canal del Plan de Marketing. Fuente: Elaboración propia. 80

Tabla 15 Plantilla propuesta para los Reportes de Marketing 1, 2 y 3. Fuente: Elaboración propia. 82
Tabla 16 Síntesis de las condiciones que motivan la propuesta, contrastando lo que se mantiene del marco vigente con lo que se adapta en esta propuesta. Fuente: Elaboración propia. 84
Tabla 17 Observaciones del 3IE y ajustes incorporados en la propuesta. Fuente: Elaboración propia. 88
Tabla 18 Secciones de la encuesta y su objetivo de medición. Fuente: Elaboración propia. 90
Tabla 19 Criterio de validación de cada ajuste. Fuente: Elaboración propia. 91
Tabla 20 Participantes de la encuesta según rol, campus y naturaleza del proyecto. Fuente: Elaboración propia. 92
Tabla 21 Porcentaje de acuerdo por afirmación, para el total de participantes y por rol. Fuente: Elaboración propia. 96
Tabla 22 Síntesis de la validación de cada propuesta. Fuente: Elaboración propia. 100
Tabla 23 Cambios de la propuesta respecto del marco vigente y de la práctica actual del curso. Fuente: Elaboración propia. 111

INTRODUCCIÓN

La Feria de Software es el ciclo formado por Gestión de Proyectos de Informática (GPI) y Taller de Desarrollo de Proyectos de Informática (TDPI), las dos asignaturas con que la carrera de Ingeniería Civil Informática de la UTFSM cierra la formación de sus estudiantes. Durante dos semestres, equipos de estudiantes identifican un problema, proponen una solución innovadora y la desarrollan como un proyecto de software, que culmina en un evento abierto a la comunidad. Este trabajo se organiza con un marco basado en Scrum definido en 2019 [1], que desde entonces se ha complementado con un enfoque de innovación para las primeras semanas de GPI [2].

Las condiciones bajo las que se desarrollan los proyectos han cambiado desde que ese marco se definió. Hoy una parte importante de los proyectos se centra en un modelo de Machine Learning, cuyo trabajo es exploratorio y no siempre deja un incremento visible al cierre de cada sprint. Los equipos usan herramientas de inteligencia artificial para programar, lo que acelera el desarrollo, pero también hace que muchas decisiones de diseño queden implícitas en el código. A esto se suma que, para el nuevo plan de estudios, que aún no entra en vigencia, se solicitó desde la Dirección de Docencia que las ingenierías civiles cuenten con una asignatura exclusiva de innovación y emprendimiento, la que en el caso de Ingeniería Civil Informática se absorberá a través de la Feria de Software[1]. Sin embargo, el emprendimiento no tiene hoy un proceso definido a lo largo de ambas asignaturas. El marco vigente no contempla ninguna de estas condiciones.

Este trabajo propone una adecuación del flujo de procesos de la Feria de Software a esta realidad. No busca reemplazar el marco de 2019, sino conservar su estructura, sus roles y sus ceremonias, y ajustar solo los puntos donde hoy se produce un desajuste. La propuesta organiza el año en procesos por semestre y agrega una clasificación de los proyectos según su naturaleza, que define si siguen el camino de los proyectos sin entrenamiento de modelo propio o el de los proyectos con entrenamiento y evaluación de modelo propio, que agrega al desarrollo de software las actividades del ciclo CRISP-DM. Además, reemplaza el Plan de Proyecto por un Documento de Diseño que se actualiza en cada sprint, e incorpora un proceso de emprendimiento con un modelo de negocios consolidado, un Plan de Marketing y tres hitos de seguimiento.

Para construir la propuesta se revisó el marco de 2019 y el trabajo de Cayo [2], junto con literatura sobre cursos Capstone, metodologías ágiles, modelos de proceso para proyectos de datos, innovación, emprendimiento y uso de inteligencia artificial en el desarrollo de software. Este análisis se complementó con la experiencia de la autora como estudiante y como ayudante de ambas asignaturas, y con un levantamiento de los proyectos presentados en la Feria de Software entre 2018 y 2025. Cada proceso propuesto se documentó con un mismo formato, que incluye su objetivo, su flujo en notación BPMN, sus entradas y salidas,

[1] Información entregada por la jefatura de carrera de Ingeniería Civil Informática.

sus reglas de negocio y una descripción, junto con plantillas para los entregables nuevos o modificados. La propuesta se validó en dos instancias, una revisión de su componente de innovación y emprendimiento con un especialista del Instituto Internacional para la Innovación Empresarial (3IE) de la UTFSM, y una encuesta a profesores, ayudantes y estudiantes de GPI y TDPI.

El documento se organiza en cinco capítulos. El capítulo 1 presenta el contexto de la Feria de Software, la situación actual del marco vigente, los objetivos y el alcance del trabajo. El capítulo 2 reúne el marco conceptual sobre el que se apoya la propuesta y cierra con una síntesis de la problemática, en la que se identifican los desajustes del marco vigente. El capítulo 3 desarrolla la propuesta de adecuación, proceso por proceso, para ambos semestres. El capítulo 4 presenta la validación de la propuesta y sus resultados. Finalmente, el capítulo 5 expone las conclusiones del trabajo, sus limitaciones y las líneas de trabajo futuro.

CAPÍTULO 1
DEFINICIÓN DEL PROBLEMA

1.1. CONTEXTO

La formación de ingenieros e ingenieras civiles en informática busca que, al finalizar sus estudios, quienes egresan sean capaces de enfrentar problemas reales de la industria, no solo desde el conocimiento técnico, sino también desde la gestión de proyectos, el trabajo en equipo y la comunicación con un cliente, estos son atributos que marcos internacionales de acreditación en ingeniería exigen explícitamente junto con la capacidad de resolver problemas complejos de ingeniería [4]. Para lograr este objetivo, distintas instituciones a nivel mundial incorporan dentro de sus planes de estudio los llamados cursos Capstone, definidos en la literatura como la instancia de un plan de estudios de ingeniería, en la que un equipo de estudiantes integra y aplica los conocimientos adquiridos a lo largo de la carrera para resolver, de principio a fin, un problema real de la industria, replicando en la medida de lo posible las condiciones de un entorno laboral profesional [5], [6], [7]. Para el caso de la carrera de Informática, esto se traduce en llevar un proyecto de software desde la identificación de un problema hasta la entrega de un producto funcional.

Dado que este tipo de proyectos busca acercar al estudiante a la vida laboral, en el caso de Ingeniería Civil Informática es habitual que se apoyen en metodologías ágiles como marco de trabajo, ya que estas se han posicionado como el estándar predominante en la industria del desarrollo de software [8]. Esto no ocurre en todos los cursos Capstone. En otras especialidades de ingeniería, estos cursos se organizan principalmente en torno a proyectos de diseño propios de cada disciplina [5]. Optar por un marco ágil responde, entonces, a que el objetivo de la asignatura es replicar la forma en que efectivamente se trabaja en la industria del software, que es donde se desempeñarán sus egresados.

Sin embargo, la incorporación de metodologías ágiles a un contexto académico no es directa. A primera vista, la situación de un estudiante que reparte su tiempo entre el proyecto y el resto de sus asignaturas podría parecer similar a la de un profesional que participa en varias células o equipos de trabajo a la vez, algo habitual en la industria. La diferencia está en quién coordina esa carga. En una organización, es la misma empresa la que asigna a una persona a distintos equipos, conoce sus prioridades y puede ajustarlas cuando chocan entre sí, y la jornada laboral completa de esa persona está dedicada a esos proyectos. Un estudiante, en cambio, responde a asignaturas independientes entre sí, cada una con sus propios plazos y evaluaciones, sin que nadie coordine su carga total. A esto se suma que el proyecto tiene como plazo fijo un semestre o año académico, independiente de su avance real. Esta tensión entre el marco ágil y las restricciones propias del entorno universitario ha sido documentada en distintos contextos educacionales, y constituye uno de los principales desafíos al momento de diseñar este tipo de asignaturas [6].

En la Universidad Técnica Federico Santa María (UTFSM), la carrera de Ingeniería Civil Informática incorpora este tipo de experiencia a través de dos asignaturas consecutivas: «Gestión de Proyectos de Informática» (GPI) y «Taller de Desarrollo de Proyectos de Informática» (TDPI), conocidas en conjunto como los ramos de Feria de Software (FESW). En el nuevo plan de estudios de la carrera, que aún no entra en vigencia[2], ambas asignaturas se ubican en los dos últimos semestres de la carrera (IX y X semestre respectivamente). Esta ubicación no es un detalle menor, al ser de las últimas asignaturas que cursa un estudiante antes de egresar, la Feria de Software concentra gran parte de la responsabilidad de consolidar las competencias asociadas al emprendimiento, innovación, desarrollo, trabajo en equipo y gestión aprendidas a lo largo de la carrera en un solo proyecto.

1.1.1. FERIA DE SOFTWARE

Se conoce como Feria de Software (FESW) tanto al ciclo formado por las asignaturas GPI y TDPI como al evento con que este culmina, en el que los proyectos se exponen al público, a patrocinadores y a auspiciadores. Durante este ciclo, equipos de entre cuatro a seis estudiantes de la carrera deben identificar un problema complejo de ingeniería y proponer una solución innovadora, que se materializa en un proyecto de software propio, ya sea para resolver una problemática real o aprovechar una oportunidad existente. Se entiende por problema complejo de ingeniería aquel que exige la aplicación de conocimiento avanzado y presenta al menos una de las características de amplitud, conflicto entre requisitos, ausencia de una solución obvia o múltiples partes interesadas involucradas [4]. En lo posible, esta solución se enmarca dentro de los Objetivos de Desarrollo Sostenible (ODS)[3]. El proyecto se desarrolla de manera incremental a lo largo de dos semestres, trabajando con un cliente externo real. Los equipos son acompañados durante todo el proceso por profesores y por ayudantes, quienes cumplen un rol fundamental apoyando y haciendo seguimiento a los equipos. El ciclo culmina en un evento de exposición abierto a la comunidad, en el que los proyectos son presentados y evaluados por profesores y ayudantes, y otra a cargo de un panel de jurados externos a la asignatura, en una instancia conocida como Demoday.

A diferencia de una asignatura tradicional, en la Feria de Software no existe un contenido fijo que se transmita de forma expositiva: el aprendizaje ocurre principalmente a través de la ejecución del propio proyecto, es decir, aplicando el principio de «aprender haciendo» que describe la teoría del aprendizaje experiencial de Kolb [10], guiado por un marco metodológico que en la actualidad se basa en SCRUM. Esto implica que gran parte del valor formativo de la asignatura depende de que dicho marco esté correctamente adaptado a las condiciones reales bajo las cuales los estudiantes desarrollan sus proyectos, condiciones que como se detalla en la sección siguiente, han cambiado de forma relevante desde que este marco fue definido.

[2] Malla curricular de Ingeniería Civil Informática, Departamento de Informática, UTFSM, 2026.

[3] Los ODS corresponden a la agenda de 17 objetivos adoptada por las Naciones Unidas en 2015 [9].

La Feria de Software ha crecido de forma sostenida en los últimos años: de 25 proyectos en 2022, la cifra subió a 28 en 2023 y a 31 en 2024, hasta alcanzar 35 proyectos en 2025 (ver Tabla 2) [11]. Estas cifras consideran los campus Casa Central y San Joaquín, ya que la Feria de Software es un único evento para ambos[4].

Los proyectos se organizan en categorías asociadas a los mismos ODS; en la edición 2025, por ejemplo, se definieron tres categorías: industria, educación y salud[5]. Dentro de estas categorías, un número creciente de proyectos usa la inteligencia artificial como parte central de su solución: en esa misma edición se presentaron proyectos que detectan enfermedades mediante redes neuronales, que reconstruyen patrimonio cultural con IA generativa y que gestionan recursos combinando aprendizaje automático con imágenes satelitales [12].

El desarrollo del proyecto a lo largo de estos dos semestres se organiza, según el marco metodológico vigente, en diez etapas agrupadas en cuatro categorías, que se detallan a continuación junto con el resto del marco de referencia de 2019; su desarrollo en profundidad, con los hitos propios de cada etapa y los roles definidos para profesores, ayudantes y equipos, se presenta en el marco conceptual (Capítulo 2).

1.2. SITUACIÓN ACTUAL

El marco de referencia utilizado actualmente en GPI y TDPI proviene de un trabajo de título en 2019, que diagnosticó la ejecución de ambas asignaturas entre los años 2014 y 2018, donde se propuso una estructura basada en SCRUM para ordenar su desarrollo [1]. Dicha propuesta organiza el ciclo anual en diez etapas [1], agrupadas en cuatro categorías. Las dos primeras categorías ocurren durante el primer semestre (GPI): la Postulación de Proyectos comprende tres etapas, que son el llamado a postulación, la definición inicial del proyecto y la defensa de la idea ante un jurado. La Realización del Prototipo comprende dos etapas, el desarrollo de un prototipo inicial y la planificación tentativa del segundo semestre. Las otras dos categorías ocurren durante el segundo semestre (TDPI). El Desarrollo del Producto se compone de tres etapas, una por cada sprint de desarrollo, acompañadas de actividades de marketing en paralelo. Finalmente, la Exposición y Cierre de Proyectos comprende dos etapas, el evento de la Feria de Software y un período de cierre. Cada una de estas diez etapas cuenta con sus propios hitos y entregables, que se detallan en el capítulo 2 (ver Figura 2).

En cuanto a roles, Calvo [1] plantea la asignatura como si fuera una gran empresa, llamada FESW Ltda., en la que cada equipo de alumnos actúa como una pequeña empresa que postula para que se financie su proyecto. A partir de esta analogía define tres roles. Los profesores y ayudantes actúan como Stakeholders durante las entregas, es decir, como la directiva de la empresa FESW, que evalúa el avance del proyecto en cada Sprint Review. Durante las clases, los profesores cumplen además el rol de Coach, orientando a los equipos.

[4] Información sacada de la página oficial de la feria de software, https://www.feriadesoftware.cl/.
[5] Información otorgada por la Jefatura de Carrera de Ingeniería Civil Informática, USM Casa Central.

A los ayudantes se les asigna también el rol de Jefe de proyectos, que apoya a cada equipo durante los sprints, controla su avance y ayuda a resolver sus impedimentos. Dentro de cada equipo de alumnos, en cambio, se mantienen los roles que define la Guía Scrum [13]. Un Scrum Master se encarga de la gestión del equipo, y un Product Owner es responsable de contactar al cliente o usuario representativo y de cuidar que el producto se desarrolle en función del valor que este le otorga [1]. Esta distribución busca corregir una inconsistencia de la estructura que existía antes de 2019, en la que el rol de Product Owner recaía formalmente en los profesores y en los ayudantes, a pesar de que en la práctica el contacto directo con el cliente lo realizaban los propios alumnos [1].

Esta estructura ha sido, desde entonces, la principal referencia utilizada para ordenar la ejecución de ambas asignaturas. En 2025, un segundo trabajo de título revisó específicamente el componente de innovación dentro de este marco. Cayo [2] constató que la innovación no es una exigencia nueva en GPI, se evalúa desde al menos 2017, cuando los equipos deben postular su proyecto describiendo qué lo diferencia de al menos tres soluciones similares existentes en el mercado. Sin embargo, ese criterio se mantenía como un ítem más dentro de la evaluación, sin una etapa que guiara a los equipos, por lo que Cayo propuso incorporar en las primeras semanas de GPI cuatro fases de innovación (identificación de problemas y oportunidades, identificación del cliente, ideación y propuesta de valor), apoyadas en Design Thinking. Su propuesta aborda la innovación en la etapa temprana del proyecto, pero no se extiende al emprendimiento a lo largo de ambas asignaturas, uno de los puntos que aborda este trabajo.

A la fecha, la estructura de diez etapas y los roles definidos en 2019 continúan siendo, a grandes rasgos, el marco utilizado en GPI y TDPI. No obstante, existen al menos tres factores que no fueron contemplados en su formulación original. Dos de ellos ya forman parte de la realidad actual de la Feria de Software, y el tercero llegará con el nuevo plan de estudios de la carrera.

El primero es el aumento en la exigencia académica que traerá el nuevo plan de estudios, que aún no entra en vigencia. En él, GPI pasa de 5 a 7 créditos SCT y TDPI de 10 a 14 créditos SCT respecto del plan con que se formuló la propuesta de 2019[6]. El cambio no es solo de créditos. Los programas de ambas asignaturas para el nuevo plan incorporan la innovación y el emprendimiento como parte de lo que el estudiante debe lograr, incluida la competencia transversal de Innovación y Emprendimiento de la universidad, y no solo como un criterio dentro de alguna evaluación [14], [15]. Ese nivel de exigencia no existía en el marco definido en 2019. Aunque el nuevo plan todavía no se aplica, la propuesta lo considera porque el marco que se adecúe ahora será el que reciba a las primeras generaciones que cursen GPI y TDPI bajo él.

El segundo es la diversidad en la naturaleza de los proyectos. El marco de 2019 fue pensado para proyectos de desarrollo de software tradicional, pero hoy en la Feria también

[6] Créditos SCT según malla curricular de Ingeniería Civil Informática, Departamento de Informática, UTFSM, plan 73-13 (2019) y nuevo plan de estudios.

se desarrollan proyectos que entrenan su propio modelo de Machine Learning, cuyo ciclo de trabajo no necesariamente se ajusta a los mismos hitos, entradas y salidas esperadas[7]. Con la popularización de la inteligencia artificial, es esperable que una proporción creciente de equipos formule proyectos cuya propuesta de valor dependa de entrenar y evaluar un modelo. Esta tendencia ya se observa en la Feria, donde los proyectos con algún componente de inteligencia artificial pasaron de un 19% en 2018 a un 66% en 2025 (ver Tabla 2).

El tercer punto es el uso de la inteligencia artificial no como tema de proyecto sino como una herramienta dentro del propio proceso de desarrollo de software. El uso es mayoritario tanto en la industria como entre los estudiantes, un 84% de los desarrolladores encuestados por Stack Overflow en 2025 declara usar o planea usar herramientas de IA [16], cifra que en la encuesta de JetBrains del mismo año alcanza un 85% y un 62% menciona que depende de al menos un asistente o agente de IA para programar [17]. También entre estudiantes universitarios de Estados Unidos hay un 57% que reporta el uso diario o semanal de la IA para sus estudios [18]. El marco que se presentó en 2019 fue diseñado antes de que estas herramientas existieran, por lo que no contempla ningún lineamiento sobre cómo incorporar estas herramientas al ciclo de desarrollo de GPI y TDPI sin comprometer el aprendizaje de los estudiantes.

Con este trabajo no se busca reemplazar el marco anteriormente mencionado, sino adecuarlo a la realidad actual de los proyectos que se desarrollan en la Feria de Software, a las nuevas exigencias curriculares y al rol que la inteligencia artificial cumple hoy tanto en el tipo de proyectos que se desarrollan como en la forma en que se desarrollan.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Proponer una adecuación del flujo de procesos ágil utilizado en las asignaturas de Gestión de Proyectos de Informática y Taller de Desarrollo de Proyectos de Informática, que responda a la realidad actual de sus proyectos y herramientas utilizadas.

1.3.2. OBJETIVOS ESPECÍFICOS

1. Diagnosticar, mediante análisis del documento de 2019 y la experiencia como estudiante y ayudante de las asignaturas, los puntos del flujo de procesos ágil vigente que no se ajustan a las condiciones actuales de los proyectos.
2. Analizar, mediante el levantamiento de una muestra de proyectos de la Feria de Software y de estudios recientes sobre el uso de IA en el desarrollo de software, cómo

[7] Esta observación motiva el primer objetivo específico, que busca precisamente diagnosticar en qué puntos concretos este desajuste ocurre.

la diversificación de los proyectos y el uso de IA como herramienta de desarrollo han cambiado la realidad del curso, para dimensionar su impacto sobre las etapas e hitos definidos en el marco actual.
3. Identificar los modelos y ciclos de desarrollo ágil vigentes en la industria de software y las herramientas asociadas a su aplicación, para contrastarlos con el marco de referencia utilizado actualmente.
4. Proponer ajustes concretos al flujo de procesos, documentarlos en esta memoria, incorporando los hallazgos de los diagnósticos anteriores.
5. Validar la propuesta mediante una revisión de su componente de innovación y emprendimiento con un especialista del 3IE y una encuesta de retroalimentación con profesores, estudiantes y ayudantes de GPI/TDPI.

1.3.3. ALCANCE

El alcance de este trabajo incluye el diagnóstico y la propuesta de adecuación del flujo de procesos ágil utilizado en las asignaturas de Gestión de Proyectos de Informática y Taller de Desarrollo de Proyectos de Informática, considerando los campus Casa Central y San Joaquín de la UTFSM, donde ambas asignaturas siguen el mismo marco y culminan en un único evento de Feria de Software. La validación de la propuesta se realiza mediante una revisión de su componente de innovación y emprendimiento con un especialista del Instituto Internacional para la Innovación Empresarial (3IE) de la UTFSM, y una encuesta de retroalimentación aplicada a profesores, ayudantes y estudiantes de ambas asignaturas, sin contemplar la implementación de un piloto durante su desarrollo. Se eligió al 3IE porque es la unidad de la UTFSM especializada en innovación y emprendimiento, colaborando esporádicamente con la Feria de Software, por ejemplo aportando el formato de plantillas que los equipos utilizan como Lean Canvas. Quedan fuera del alcance de este trabajo el desarrollo de una herramienta de software que dé soporte o automatice los procesos propuestos, así como la implementación de la propuesta para medir sus resultados en un contexto real; ambas quedan planteadas como trabajo futuro.

1.4. METODOLOGÍA DEL TRABAJO

El trabajo se desarrolló en cinco pasos sucesivos, cada uno apoyado en el anterior.

Primero, se analizó el marco vigente definido por Calvo en 2019, con sus etapas, roles e hitos, y se revisó la literatura sobre Scrum y metodologías ágiles en la industria, cursos Capstone, innovación y emprendimiento, uso de IA en el desarrollo, ciclo de vida de proyectos de Machine Learning y tratamiento de datos personales (capítulo 2).

Segundo, se revisaron los proyectos publicados en la Feria de Software entre 2018 y 2025, para dimensionar cómo ha cambiado el tipo de proyecto que llega al curso. La experiencia

de la autora como estudiante y ayudante de GPI y TDPI se incorporó como contexto complementario, sin considerarla evidencia por sí sola.

Tercero, a partir del contraste entre el marco vigente, la literatura y la revisión de proyectos, se identificaron los desajustes entre el flujo actual y la realidad del curso (capítulo 2).

Cuarto, se construyó la propuesta de adecuación, conservando la estructura del marco vigente y ajustando solo los puntos donde se identificó un desajuste (capítulo 3).

Por último, la propuesta se validó en dos instancias, una revisión de su componente de innovación y emprendimiento con el 3IE, cuyas observaciones se incorporaron antes de continuar, y una encuesta a profesores, ayudantes y estudiantes de ambas asignaturas (capítulo 4).

CAPÍTULO 2
MARCO CONCEPTUAL

2.1. GESTIÓN DE PROYECTOS Y METODOLOGÍAS DE DESARROLLO DE SOFTWARE

Antes de abordar los conceptos específicos de la propuesta, es necesario establecer una base común de vocabulario. Esta sección recorre niveles progresivamente más específicos: primero la disciplina general de la gestión de proyectos, luego el contraste entre metodologías tradicionales y ágiles que explica por qué la industria del software migró hacia estas últimas, después Scrum, el marco ágil concreto sobre el que se construye el flujo de procesos vigente en GPI y TDPI, cuyos roles, eventos y artefactos se retoman constantemente a lo largo de este trabajo. Finalmente, presenta un panorama de otros marcos y herramientas ágiles vigentes en la industria, que sitúa a Scrum dentro de un ecosistema más amplio.

2.1.1. GESTIÓN DE PROYECTOS

Un proyecto se define como un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único, dentro de restricciones de tiempo, costo, alcance y calidad determinadas [19]. La gestión de proyectos corresponde a la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con sus requisitos [19]. A diferencia de las actividades habituales de una organización, que son continuas y repetitivas, un proyecto tiene un inicio y un fin definidos y produce un resultado distinto cada vez, lo que exige una gestión activa de sus riesgos, su alcance y los recursos disponibles para llevarlo a cabo.

Las asignaturas de GPI y TDPI buscan replicar esta misma lógica dentro de un contexto académico, donde cada equipo de estudiantes debe gestionar su proyecto de software bajo restricciones de tiempo, acotadas al calendario académico, de alcance, definido por el propio equipo al postular una solución a un problema real y de recursos, ya que se encuentran limitados a la disponibilidad horaria de sus integrantes y un presupuesto acotado si no tienen un cliente que los patrocine. En este sentido, el conjunto de estas asignaturas no solo evalúa la capacidad técnica de los estudiantes para construir un producto de software, sino también su capacidad de gestionar un proyecto real bajo condiciones de incertidumbre, con hitos y entregables definidos, tal como ocurriría en un contexto profesional. Es por esto que es fundamental que este marco conceptual parta precisamente por definir qué se entiende por gestión de proyectos, antes de abordar la metodología específica utilizada para llevarla a cabo.

2.1.2. METODOLOGÍAS DE DESARROLLO: TRADICIONALES VS. ÁGILES

Dentro de las metodologías tradicionales de desarrollo de software está la llamada metodología en cascada o waterfall, que organiza el ciclo de vida de un proyecto en fases secuenciales y delimitadas (levantamiento de requisitos, diseño, implementación, verificación y mantención), donde cada fase debe completarse y validarse antes de dar inicio a la siguiente [20]. Este enfoque asume que es posible definir la totalidad de los requisitos de un producto antes de comenzar su construcción, lo que permite una planificación detallada y una asignación de recursos predecible desde el inicio del proyecto. Sin embargo, esta rigidez se convierte en una limitación cuando los requisitos cambian durante el desarrollo, algo que es frecuente en los proyectos de software. Esto implica que incorporar un cambio requiere retroceder a fases que ya han sido cerradas y un aumento en el costo de tiempo y recursos.

Como respuesta a estas limitaciones, un grupo de desarrolladores de software se reunió en 2001 para redactar el Manifiesto por el Desarrollo Ágil de Software, documento que sintetiza valores y principios orientados a favorecer la entrega temprana y continua de software funcional por sobre documentación, que si bien es necesaria para el desarrollo de software, no es el objetivo final del mismo [21]. A diferencia de las metodologías tradicionales, los enfoques ágiles dividen el desarrollo en ciclos cortos e iterativos, llamados Sprints o iteraciones, al término de los cuales se entrega un incremento funcional del producto que puede ser evaluado y ajustado antes de continuar. Bajo este esquema, los requisitos no se fijan por completo al inicio del proyecto, sino que se van refinando y priorizando a medida que el equipo y el cliente aprenden más sobre el producto que se está construyendo, lo que hace de lo ágil un enfoque particularmente adecuado para contextos de alta incertidumbre, donde no es posible, o no conviene, anticipar todos los requisitos desde el comienzo.

Esta capacidad de adaptarse al cambio ha llevado a que las metodologías ágiles se posicionen como el estándar en la industria de desarrollo de software actual, desplazando en una gran medida a los enfoques tradicionales en proyectos donde la incertidumbre y el cambio de requisitos son la norma más que una excepción [8]. Dentro de las metodologías ágiles existen múltiples marcos de trabajo con énfasis distintos, como se detalla más adelante en esta sección, donde Scrum es el que se ha adoptado como marco de trabajo específico en las asignaturas de GPI y TDPI, por lo que se aborda a detalle a continuación.

2.1.3. SCRUM

Scrum se define, según la Guía Scrum en su versión 2020, como un marco de trabajo ligero que ayuda a personas, equipos y organizaciones a generar valor a través de soluciones adaptativas para problemas complejos [13]. A diferencia de una metodología prescriptiva, que detalla paso a paso cómo ejecutar cada actividad, Scrum se presenta como un marco de trabajo ágil, que define un conjunto mínimo de roles, eventos, artefactos y reglas para estructurar la colaboración de un equipo, el que tiene la responsabilidad de definir

los procesos y técnicas específicas con las que llevará a cabo su trabajo. Este marco se sustenta en los pilares del empirismo (transparencia, inspección y adaptación) y en los valores del pensamiento lean, que busca concentrar el esfuerzo del equipo en aquello que efectivamente genera valor. Su propósito no es entregar una solución completa desde el inicio, sino avanzar de forma iterativa e incremental, entregando en cada ciclo un incremento del producto que pueda ser revisado y ajustado antes de continuar, lo que permite responder a la incertidumbre propia de los proyectos de software sin necesidad de anticipar todos los requisitos desde un comienzo.

La Guía Scrum define tres roles dentro de lo que en su versión de 2020 se denomina el Scrum Team: el Product Owner, el Scrum Master y los Developers. El Product Owner es responsable de maximizar el valor del producto resultante del trabajo del equipo, principalmente a través de la gestión del Product Backlog: define y prioriza qué se construye y en qué orden, actuando como el vínculo principal entre el equipo y sus interesados, incluyendo al cliente o usuario del producto. El Scrum Master es responsable de establecer Scrum tal como se define en la Guía, ayudando a todos los involucrados a entender y aplicar la teoría y las prácticas del marco, tanto dentro del equipo como en la organización en la que se inserta. A diferencia de un jefe de proyecto tradicional, no dirige el trabajo del equipo, sino que facilita su autogestión y elimina los impedimentos que puedan surgir. Los Developers, finalmente, son quienes se comprometen a crear cualquier aspecto de un incremento utilizable en cada Sprint, y son responsables de la calidad técnica del producto y de la planificación propia del trabajo necesario para cumplir con el objetivo del Sprint [13].

Los eventos de Scrum se organizan en torno al Sprint, usualmente de una a cuatro semanas, tiempo en donde se busca generar un incremento de un producto utilizable y ocurren los eventos descritos a continuación. Al inicio de cada Sprint se realiza la Sprint Planning, momento donde el Scrum Team define el objetivo del Sprint y selecciona desde el Product Backlog el trabajo que se compromete a realizar durante ese periodo. Durante el desarrollo del Sprint, el equipo se reúne diariamente en el Daily Scrum, un evento breve, de no más de quince minutos, en el que los Developers comentan su avance dirigido hacia el objetivo del Sprint y ajustan su plan de trabajo de los días siguientes. Al cierre del Sprint se realizan dos eventos consecutivos: la Sprint Review, que consiste en que el equipo presenta el incremento desarrollado a sus interesados para obtener una retroalimentación y ajustar el Product Backlog en consecuencia, y la Sprint Retrospective, donde el equipo evalúa cómo fue su propio funcionamiento durante el Sprint, en cuanto a personas, procesos y herramientas, identificando mejoras a implementar en el siguiente ciclo [13].

Los artefactos de Scrum representan el trabajo o el valor generado por el equipo y cada uno de ellos aumenta la transparencia del proceso y permite medir el progreso. El Product Backlog es una lista ordenada y emergente de todo lo que se necesita para mejorar el producto, gestionada y priorizada por el Product Owner, y se orienta a un objetivo de largo plazo, llamado Product Goal, que da coherencia y dirección al trabajo del equipo a lo largo de los distintos Sprints. El Sprint Backlog reúne el objetivo del Sprint con los elementos del

Product Backlog seleccionados para ese ciclo y el plan que el equipo define para entregarlos. Finalmente, el «Incremento» corresponde a un paso concreto hacia el Product Goal, formado por cada elemento del Product Backlog completado durante un Sprint sumado a los incrementos anteriores, su compromiso es la «Definition of Done», una descripción formal del estado que debe alcanzar un incremento para ser considerado completo, que asegura que todos los involucrados comparten un mismo entendimiento de lo que implica que un trabajo esté terminado [13].

2.1.4. ESTIMACIÓN Y SPIKES EN SCRUM

La Guía Scrum define los artefactos y eventos del marco, pero deja deliberadamente abierto cómo un equipo estima el esfuerzo de su trabajo. Dos prácticas que no forman parte de la Guía en sí, pero que se han vuelto un estándar en su implementación, y que la propuesta utiliza como base para el diseño de los sprints del segundo semestre, son los puntos de historia y los Spikes.

Los puntos de historia (story points) son una unidad de medida relativa que expresa el tamaño total de una historia de usuario o tarea, entendida como una combinación del esfuerzo, la complejidad, el riesgo y la incertidumbre que involucra su desarrollo, y no como una medida directa de tiempo. No existe una fórmula fija para asignar este valor a una historia, su magnitud solo es relevante para el equipo que la estima, generalmente mediante una técnica como el Planning Poker, que combina la opinión de expertos con el juicio por analogía frente a historias ya estimadas previamente. En GPI y TDPI, como convención operativa del curso, esta unidad se ha calibrado a una equivalencia fija de seis horas de esfuerzo por punto de historia, lo que permite traducir la carga de trabajo semanal disponible de cada integrante en una cantidad de puntos de historia que el equipo puede comprometerse a completar en cada sprint [22]. Esta equivalencia no es una característica de Scrum ni del concepto de punto de historia en general, sino un acuerdo propio del curso para facilitar la planificación de los sprints.

Un Spike, por su parte, es una actividad de investigación acotada en el tiempo que genera el conocimiento necesario para reducir la incertidumbre técnica o funcional de una historia, mediante prototipos, pruebas de concepto o investigación de viabilidad. A diferencia de una historia de usuario convencional, no produce una funcionalidad directamente utilizable por el cliente, pero sí entrega la información necesaria para tomar decisiones y continuar el desarrollo con mayor claridad. En su uso habitual dentro de GPI y TDPI, un Spike es un ítem acotado del backlog y no una etapa completa del proyecto. Los equipos pueden realizar un máximo de dos Spikes durante todo el proyecto, únicamente durante los Sprints 1 y 2 del segundo semestre, con una estimación de esfuerzo máxima de 30 puntos de historia por Spike [22]. La propuesta retoma el concepto para nombrar la actividad central del Sprint 0 en proyectos con modelo propio, pero, como se detalla en el capítulo siguiente, extiende deliberadamente su alcance habitual de un ítem acotado dentro de un sprint a una etapa

completa, dada la magnitud de la incertidumbre técnica que un proyecto de Machine Learning enfrenta en esta etapa.

2.1.5. METODOLOGÍAS Y HERRAMIENTAS ÁGILES EN LA INDUSTRIA

Aunque Scrum es el marco de trabajo utilizado en GPI y TDPI, no es el único enfoque ágil vigente en la industria del software. Kanban, por ejemplo, organiza el trabajo como un flujo continuo visualizado en un tablero, limitando el trabajo en progreso para optimizar el rendimiento, en lugar de organizarlo en ciclos fijos como los Sprints de Scrum. Extreme Programming (XP) enfatiza prácticas técnicas como la integración continua y la programación en pares para maximizar la velocidad y la calidad del código, mientras que marcos como Scaled Agile Framework (SAFe) extienden los principios ágiles a organizaciones grandes que requieren coordinar múltiples equipos en un mismo producto. Un estudio comparativo reciente de estos cuatro marcos concluye que cada uno resuelve un problema distinto: Kanban destaca en eficiencia de flujo, XP en velocidad y excelencia técnica, Scrum en colaboración con los interesados dentro de entornos cambiantes, y SAFe en la coordinación de iniciativas complejas a gran escala [23].

El informe State of Agile, la encuesta de referencia más consultada sobre adopción de metodologías ágiles en la industria, señala una tendencia hacia la diversificación: el 74% de las organizaciones encuestadas hoy utiliza modelos híbridos o adaptados internamente, en lugar de seguir un marco formal sin modificaciones [24]. Esto refuerza que la elección de un marco de trabajo particular no es un fin en sí mismo, sino una decisión que depende del contexto de cada equipo u organización, tal como se plantea en la sección *Adaptación de marcos ágiles* respecto a la adecuación de Scrum al contexto de la Feria de Software.

En cuanto a herramientas, el mismo informe reporta que un 54% de las organizaciones utiliza software de planificación ágil empresarial (como Jira o Azure DevOps) para dar seguimiento a su trabajo, y que un 41% ya explora o está implementando activamente herramientas de inteligencia artificial dentro de su flujo de desarrollo, principalmente asistentes de código como GitHub Copilot (57%) y modelos de lenguaje de propósito general (65%) [24].

En el caso específico de GPI y TDPI, Calvo [1] evaluó en el marco teórico de su memoria de 2019 tanto Scrum como Extreme Programming (XP) y el desarrollo a medida (ad-hoc) para el contexto de la Feria de Software, descartando estas dos últimas alternativas: XP exige un nivel de dedicación (40 horas semanales, cliente presente de forma constante) y de coaching por parte de los instructores que no es viable en un curso donde los estudiantes compatibilizan el proyecto con el resto de sus asignaturas, mientras que el desarrollo ad-hoc no entrega ninguna estructura que oriente a equipos con poca experiencia. Sin embargo, la elección final de Scrum como base de la propuesta que hoy sigue vigente no respondió a esa comparación teórica, sino a un criterio más práctico planteado en el capítulo de propuesta de la misma memoria: los estudiantes ya llegan a la asignatura con conocimientos básicos de Scrum adquiridos en cursos previos y además, este es el marco ágil más popular en Chile

[1]. Dado que el contexto educacional impide implementar Scrum «al pie de la letra», ya que sus hitos y normas exigen un cumplimiento estricto que no siempre es compatible con las restricciones de un curso académico, Calvo definió su propuesta explícitamente como «basada en Scrum» y no como una implementación pura del framework, complementándola con otras herramientas y técnicas que facilitaran su aplicación en las asignaturas [1]. Esta distinción entre un framework «puro» y uno «basado en» resulta relevante para este trabajo, ya que la misma lógica de adecuación de tomar Scrum como punto de partida y ajustarlo a las condiciones reales del proyecto, en lugar de descartarlo o aplicarlo sin modificaciones, es la que se retoma en la propuesta que se desarrolla en el capítulo siguiente. Contrastado con el panorama actual de la industria, el marco de Calvo coincide en lo esencial, ya que también es una adaptación de Scrum, en línea con el 74% de organizaciones que hoy usan modelos híbridos o adaptados [24]. Sin embargo, fue definido antes de que herramientas de inteligencia artificial como las que hoy explora el 41% de las organizaciones formaran parte del flujo de desarrollo, una diferencia que la propuesta aborda.

2.2. CURSOS CAPSTONE Y APRENDIZAJE BASADO EN PROYECTOS

El Aprendizaje Basado en Proyectos (ABP) es una metodología pedagógica que organiza el aprendizaje en torno al desarrollo de un proyecto extendido en el tiempo, en el que los estudiantes investigan y responden a una pregunta o problema complejo y auténtico [25]. A diferencia de la enseñanza tradicional, en la que el conocimiento se transmite de manera secuencial y el estudiante lo recibe de forma pasiva, el ABP invierte esta lógica. El conocimiento se construye como consecuencia de la necesidad de resolver un problema concreto, en un proceso de investigación guiado por el propio estudiante.

En el contexto específico de la formación en ingeniería, el ABP se ha consolidado como una de las metodologías más utilizadas para cerrar la brecha entre la formación académica y las competencias que exige el mundo profesional. La literatura identifica tres tipos de aportes que esta metodología entrega a los estudiantes de ingeniería: el desarrollo de competencias técnicas, personales y contextuales de forma simultánea, el trabajo con problemas reales del ámbito profesional en lugar de ejercicios acotados y artificiales, y el fomento del aprendizaje colaborativo, al requerir que los estudiantes trabajen en equipo para resolver un problema que ninguno podría abordar individualmente [26].

El curso Capstone corresponde a la aplicación de estos principios en su forma más completa y exigente dentro de un programa de ingeniería: una asignatura o secuencia de asignaturas, ubicada al final de la carrera, en la que un equipo de estudiantes debe desarrollar un proyecto de principio a fin, integrando y poniendo a prueba los conocimientos y competencias adquiridos a lo largo de toda su formación. Una revisión sistemática reciente de 127 cursos Capstone de ingeniería de software identifica un conjunto de características comunes a este tipo de asignaturas más allá de las diferencias entre programas: la conformación de equipos de trabajo, la existencia de un cliente o interesado externo, real o

simulado, que define el problema a resolver, la entrega de un producto de software funcional al término del curso, y una evaluación centrada en el resultado y el proceso seguido para alcanzarlo, más que en el dominio de contenidos específicos [27].

La Feria de Software corresponde precisamente a este tipo de instancia: un curso Capstone que busca consolidar, en un solo ejercicio de dos semestres, las competencias técnicas y de gestión desarrolladas a lo largo de la carrera. Sin embargo, como se detalla en la sección siguiente, la literatura actual dice que llevar un marco de trabajo ágil, pensado originalmente para equipos profesionales con dedicación completa, a un contexto de esta naturaleza no está libre de tensiones y desajustes.

2.2.1. METODOLOGÍAS ÁGILES Y RESTRICCIONES DEL ENTORNO ACADÉMICO

La adopción de metodologías ágiles, y en particular de Scrum, en cursos Capstone es hoy una práctica común y documentada en múltiples instituciones [27], [28]. Sin embargo, la literatura es consistente en señalar que trasladar un marco ágil, diseñado originalmente para equipos profesionales, a un contexto universitario no es algo directo, ya que genera fricciones con la estructura del marco que deben ser gestionadas o adaptadas específicamente al contexto [6], [29].

La primera fricción documentada es la dedicación. Un equipo profesional cuenta con dedicación exclusiva al proyecto durante la jornada laboral, mientras que un equipo de estudiantes debe compatibilizar su proyecto con la carga del resto de sus asignaturas; esta diferencia no es solo en horas disponibles, sino que afecta directamente la coordinación del equipo. El estudio de Masood, Hoda y Blincoe [29] en la Universidad de Auckland constató que encontrar horarios comunes para que todo el equipo trabajara de forma sincrónica fue una de las principales dificultades reportadas por los estudiantes, lo que llevó a los equipos a adaptar prácticas como el Daily Scrum, reduciendo su frecuencia respecto a lo prescrito por la Guía Scrum.

La segunda fricción es el calendario. Un equipo profesional itera hasta que el producto alcanza el nivel de madurez o un valor esperado, ajustando la duración del proyecto según sea necesario, mientras que un curso Capstone tiene una duración fija determinada por el calendario académico, independiente del avance real del proyecto [6]. Esta restricción obliga a que los Sprints e hitos del curso se definan en función del semestre o año académico y no en función del producto, lo que puede llevar a que el equipo entregue funcionalidades incompletas o de menor calidad para cumplir con una fecha.

La tercera fricción es la lógica de evaluación. En Scrum, el progreso se mide a través de la entrega continua de incrementos de valor para el cliente o usuario, mientras que un curso Capstone debe además evaluar a cada estudiante con una nota, siguiendo los lineamientos de una pauta [6]. Esto introduce un objetivo adicional, aprobar la asignatura, que no siempre está alineado con el objetivo del Scrum Team de maximizar el valor del producto,

ni necesariamente con el objetivo individual de cada estudiante de obtener una buena calificación [7]. A esta tensión entre la lógica de evaluación académica y la lógica de entrega de valor de Scrum se suma, en el caso particular de la Feria de Software, que no todo el equipo persigue necesariamente el mismo objetivo dentro del curso.

Estas tres fricciones no son exclusivas de la Feria de Software. Son consistentes con lo que la literatura documenta en otros programas de ingeniería que han adoptado marcos ágiles para sus cursos Capstone, lo que sugiere que no se trata de un problema de implementación particular de GPI y TDPI, sino de una tensión estructural inevitable al trasladar un marco pensado para la industria a un contexto académico.

2.2.2. ETAPAS Y ROLES DEL MARCO VIGENTE

Antes de 2019, GPI y TDPI se organizaban con una estructura de roles distinta, donde el ayudante tenía las responsabilidades de Product Owner y de Scrum Master, mientras que los profesores, que también figuraban como Product Owner, actuaban como una «Gerencia» que dirigía el proyecto más que lo educaba. Calvo [1] identificó esta superposición como una fuente de inconsistencias: en la práctica, el contacto directo con el cliente lo sostenían los propios alumnos, no el ayudante ni el profesor, pese a que el rol de Product Owner recaía formalmente en estos últimos.

Para corregir esta inconsistencia, la propuesta de 2019 redistribuye los roles de la forma que se mantiene vigente hasta hoy (ver Figura 1). A los profesores se les asigna un doble rol, como Stakeholders que evalúan el avance del proyecto en cada Sprint Review, y como Coach que orienta a los equipos, al ayudante se le asigna el rol de Jefe de proyectos (denominado Team Lead en la propuesta original), apoyando a los equipos desde fuera sin asumir un rol formal dentro de la estructura Scrum y dentro de cada equipo de alumnos se definen dos liderazgos internos, un Scrum Master encargado de la gestión del equipo y un Product Owner responsable del contacto con el cliente y de velar porque el desarrollo responda al valor que este le otorga al producto [1].



Figura 1: Roles de la propuesta vigente en GPI y TDPI.
Fuente: Calvo [1].

En cuanto al proceso, la propuesta organiza el ciclo anual en diez etapas (ver Figura 2), agrupadas en cuatro categorías. La categoría de Postulación de Proyectos abarca el llamado a postulación, la definición inicial del proyecto y la defensa de la idea ante un jurado. La categoría de Realización del Prototipo comprende el desarrollo de un prototipo inicial y la redacción de una planificación tentativa para el segundo semestre. La categoría de Desarrollo del Producto se compone de tres sprints: el primero de desarrollo inicial, el segundo que incorpora la retroalimentación recibida, y el tercero que incluye una revisión completa junto al cliente[8]. Finalmente, la categoría de Exposición y Cierre de Proyectos culmina con el evento de la Feria de Software y un período de cierre de proyectos, instancia que Calvo [1] incorpora como novedad respecto de lo que se hacía antes de su propuesta, pensada para canalizar de forma más efectiva la retroalimentación de los estudiantes hacia profesores y ayudantes.

[8] En la práctica actual esta secuencia no siempre se cumple: el cliente suele incorporarse antes, entre el Sprint 1 y el Sprint 2, y no recién en el Sprint 3 como establece la propuesta original.

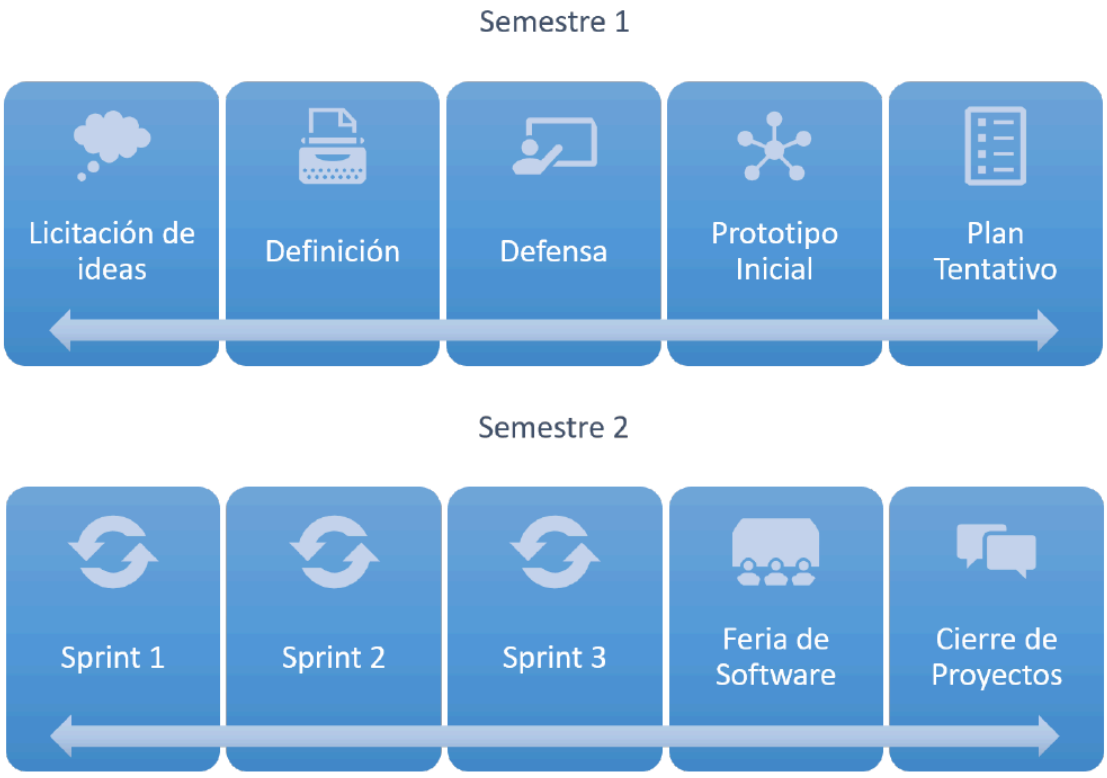


Figura 2: Las diez etapas de la propuesta vigente en GPI y TDPI, agrupadas en cuatro categorías.
Fuente: Calvo [1].

Cada una de estas diez etapas posee, a su vez, sus propios hitos y evaluaciones específicas (los entregables concretos que los equipos deben presentar dentro de cada etapa). Esta estructura de etapas y roles es la que se mantiene, a grandes rasgos, como marco de referencia vigente en la Feria de Software y es sobre la que se apoya el resto de este trabajo para identificar en qué puntos concretos ha dejado de ajustarse a la realidad actual de los proyectos.

2.3. INNOVACIÓN Y EMPRENDIMIENTO EN PROYECTOS DE SOFTWARE

El Manual de Oslo, la referencia internacional para medir la innovación, la define como la creación de un producto o proceso, nuevo o significativamente mejorado respecto a lo que existía antes [30]. Cuando ese producto o proceso se logra aplicando conocimientos y herramientas técnicas para resolver un problema o satisfacer alguna necesidad, se habla de innovación tecnológica, concepto presente en los proyectos de la Feria de Software.

Innovación y emprendimiento son conceptos relacionados, pero no equivalentes. Mientras que la innovación es el proceso de crear o mejorar un producto o proceso, el

emprendimiento corresponde al proceso de descubrir, evaluar y explotar una oportunidad para convertir esa idea en una actividad viable. Esto exige validar que existe un cliente dispuesto a pagar por la idea, que la propuesta compite favorablemente frente a las alternativas existentes y que la oportunidad tiene un tamaño de mercado que la justifique [31]. Un proyecto puede ser innovador sin llegar a ser realmente emprendedor, si nunca somete su idea a este tipo de validación.

El emprendimiento temprano corresponde específicamente a las primeras etapas de este proceso, cuando todavía no existe una empresa constituida ni un producto validado en el mercado y el objetivo no es construir la solución completa, sino confirmar que la necesidad es real y que la propuesta le interesa a un cliente [32].

Más allá de esta definición inicial, es importante entender el emprendimiento como un proceso compuesto por etapas sucesivas y no como un entregable puntual que se resuelve de una sola vez. El modelo de Customer Development propuesto por Blank [32] distingue cuatro etapas:

1. Customer Discovery, en la que el equipo valida que el problema identificado es real y relevante para un cliente.
2. Customer Validation, en la que se confirma que existe un modelo de negocio replicable y escalable en torno a la solución propuesta.
3. Customer Creation, orientada a generar demanda del producto una vez validado el modelo.
4. Company Building, en la que la iniciativa pasa de un equipo en modo de búsqueda y experimentación a una organización formal.

El emprendimiento temprano descrito anteriormente corresponde, en estos términos, únicamente a las dos primeras etapas de este proceso (Customer Discovery y Customer Validation), quedando las dos últimas fuera del alcance de un curso Capstone de dos semestres como la Feria de Software.

Esta noción de proceso es la que exige hoy el marco institucional de la UTFSM. El modelo educativo institucional define la Innovación y Emprendimiento como una de sus competencias transversales sello, entendida como la capacidad de «desarrollar mejoras e innovaciones tecnológicas y de gestión, generando oportunidades para dar respuesta satisfactoria a las necesidades organizativas y sociales» [33]. Se trata de una competencia que no se demuestra en una entrega única, sino a lo largo de un proceso formativo completo, exigencia que, como se detalla en el capítulo 1, los programas de GPI y TDPI del nuevo plan de estudios ya traducen en resultados de aprendizaje escalonados para esta competencia. La propuesta de 2019 que rige actualmente la Feria de Software, en contraste, trata ambos conceptos de forma más operativa que conceptual: define la innovación simplemente como identificar oportunidades de producto que no se encuentren en el mercado y el emprendimiento como el trabajo en equipo para sacar adelante un proyecto de software [1], sin problematizar ninguno de los dos como un proceso con etapas propias. Este trabajo retoma, en cambio, la definición más desarrollada presentada al inicio de esta sección para

evaluar en qué medida el marco vigente permite a los equipos recorrer efectivamente ese proceso.

Una de las metodologías más usadas para hacer esta validación temprana es NABC (Necesidad, Aproximación, Beneficios y Competencia), que propone responder cuatro preguntas simples para construir una propuesta de valor: ¿Qué necesidad se está resolviendo?, ¿Cuál es la solución propuesta?, ¿Qué beneficio concreto entrega esa solución frente a su costo? y ¿Cómo se compara con otras alternativas que ya existen? [34]. Como se detalla en la siguiente sección, es en esta segunda mitad, la de los beneficios frente al costo invertido y la competencia, donde el enfoque de innovación en la Feria de Software aún es débil.

2.3.1. EL ENFOQUE DE INNOVACIÓN EN LA FERIA DE SOFTWARE 2025

En su memoria de 2025, Benjamín Cayo Vidal [2] identificó que la exigencia de innovación en los proyectos de la Feria de Software no es un requisito nuevo, se evalúa desde al menos 2017 a través de la rúbrica de postulación de proyectos, en la que los equipos deben diferenciar su propuesta de al menos tres soluciones similares existentes en el mercado. Sin embargo, su diagnóstico identifica que este criterio se ha mantenido como un ítem más dentro de la evaluación general, sin que exista una etapa o actividad sistemática que guíe a los equipos en la detección de problemas y el desarrollo de ideas innovadoras.

Para cerrar esta brecha, Cayo [2] propone un marco de trabajo que incorpora metodologías y herramientas de innovación tecnológica en las etapas tempranas del proyecto, apoyado principalmente en Design Thinking y en un conjunto de técnicas de creatividad, identificación de problemas y emprendimiento, entre ellas Lean Startup y el Lienzo de Modelo de Negocio de Osterwalder [35]. Su propuesta se organiza en cuatro fases secuenciales: identificación de problemas y oportunidades, identificación del cliente, ideación, y propuesta de valor, cada una acompañada de herramientas y plantillas específicas para que los equipos las trabajen durante las primeras semanas de GPI (ver Figura 3).

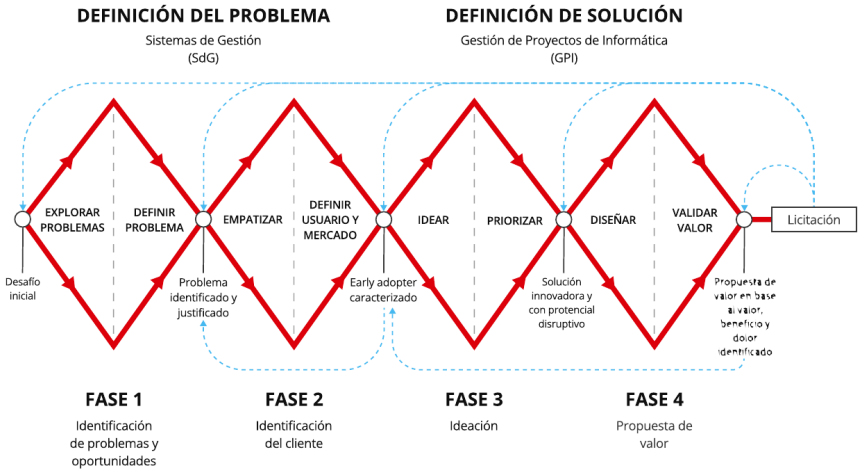


Figura 3: Modelo del marco de trabajo para la innovación tecnológica propuesto. Fuente: Cayo [2].

Cayo [2] validó su propuesta mediante un piloto realizado en GPI, cuyos resultados muestran que, si bien los objetivos de cada fase se cumplieron, el tiempo acotado disponible dentro del semestre limitó la profundidad con la que los equipos pudieron trabajar cada etapa, una restricción consistente con la fricción de calendario descrita en la sección *Metodologías ágiles y restricciones del entorno académico*. Este trabajo no busca reformular el enfoque de innovación que propone Cayo, sino tomarlo como el estado del arte vigente en esta materia dentro de la Feria de Software y construir sobre él lo que no aborda, que es el emprendimiento.

Vale la pena notar que las cuatro fases del marco de Cayo [2] llegan hasta la propuesta de valor, pero no se extienden hacia su validación como oportunidad de negocio. No contemplan una etapa dedicada a cuantificar los beneficios de la solución frente a su costo, a analizar formalmente a la competencia o a dimensionar el tamaño de la oportunidad de mercado, que es justamente la segunda mitad de NABC descrita al inicio de esta sección. Esto no es una crítica al trabajo de Cayo, que se concentra deliberadamente en la etapa más temprana del proyecto dentro de GPI, sino el punto de partida de la sección siguiente.

2.3.2. INNOVACIÓN SIN EMPRENDIMIENTO

Esta brecha no es exclusiva de la Feria de Software. La literatura también la documenta. El Design Thinking se ha vuelto una metodología central para enseñar innovación en ingeniería, pero requiere una forma de pensar distinta a la que usan los enfoques tradicionales de emprendimiento, como el análisis de mercado o los estudios de factibilidad. Por eso existe una brecha entre enseñar a innovar y enseñar a evaluar si esa innovación funciona como negocio [36]. En otras palabras, un equipo puede aprender a innovar sin nunca aprender a evaluar si su idea es un negocio viable.

Hay modelos que sí logran cerrar esta brecha. El programa I-Corps de la National Science Foundation, que ya usan decenas de universidades de ingeniería en Estados Unidos, además de pedir a los equipos identificar un problema y proponer una solución, los obliga a validar esa solución con clientes reales y a calcular si es viable como negocio, siguiendo el método de Customer Discovery [32], [37].

En la Feria de Software pasa algo parecido a lo que describe la literatura. El enfoque de innovación que se revisó en la sección anterior se trabaja en GPI, en la etapa más temprana del proyecto, pero ni GPI ni TDPI tienen hoy una etapa dedicada a validar esa idea como oportunidad de negocio: calcular el beneficio que le entrega al cliente frente a su costo, analizar a la competencia o estimar el tamaño del mercado. Agregar esta etapa una vez que el equipo demostró que puede construir su solución y antes de iniciar el desarrollo del segundo semestre acercaría la experiencia de la Feria de Software a la de un equipo real, donde no basta con tener una buena idea si esta no se traduce en un negocio viable o beneficio directo para algún posible cliente.

2.3.3. HERRAMIENTAS PARA MODELAR Y COMUNICAR UN EMPRENDIMIENTO

Para que un equipo pueda recorrer el proceso de emprendimiento descrito en esta sección, necesita herramientas que le permitan, por un lado, ordenar cómo funciona su negocio y, por otro, comunicarlo a sus clientes. La propuesta utiliza dos lienzos para lo primero y un conjunto de conceptos de marketing para lo segundo.

El Business Model Canvas, propuesto por Osterwalder y Pigneur [35], parte de la idea de que un modelo de negocio describe la lógica con que una organización crea, entrega y captura valor, y la representa en un lienzo de nueve bloques: segmentos de clientes, propuesta de valor, canales, relación con los clientes, fuentes de ingresos, recursos clave, actividades clave, socios clave y estructura de costos. Su fortaleza es que obliga al equipo a mirar el negocio completo en una sola hoja, incluyendo cómo va a operar y con quién, y no solo qué problema resuelve. En el curso, este lienzo se trabaja con el formato de la Figura 4 [38].

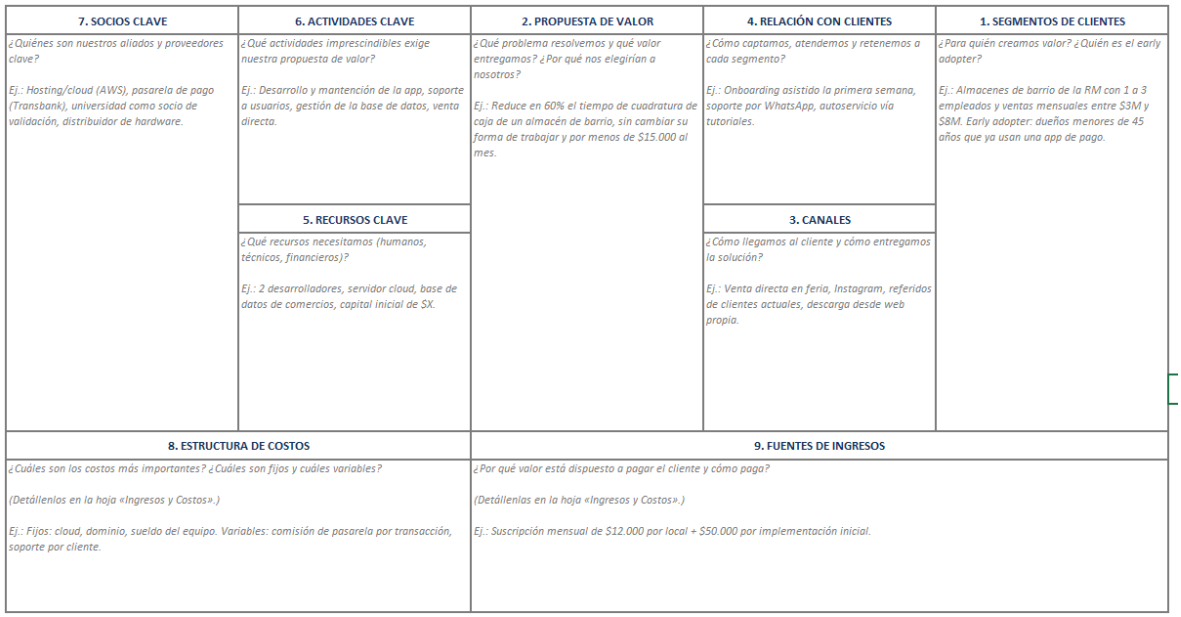


Figura 4: Formato de Business Model Canvas utilizado en TDPI. Fuente: Material de la asignatura (Actividad E7).

El Lean Canvas, propuesto por Maurya [39], es una adaptación del Business Model Canvas pensada para emprendimientos en etapas tempranas, donde la mayor incertidumbre no está en cómo operar el negocio, sino en si el problema existe y si la solución le interesa a alguien. Por eso reemplaza cuatro bloques del lienzo original (socios clave, actividades clave, recursos clave y relación con los clientes) por problema, solución, métricas clave y ventaja competitiva, y conserva los demás. En la práctica, ambos lienzos son complementarios: el Lean Canvas ayuda a validar la idea al comienzo y el Business Model Canvas a definir cómo se sostiene una vez que la idea tiene cierto respaldo[9].

Una vez definido el negocio, el equipo necesita darlo a conocer. La American Marketing Association define el marketing como las actividades y procesos destinados a crear, comunicar, entregar e intercambiar ofertas que tienen valor para clientes, socios y la sociedad, lo que deja claro que el marketing no se reduce a la publicidad. Dentro de él, la gestión de marca (branding) corresponde al proceso estratégico de construir y gestionar una marca para crear una identidad diferenciadora, generar confianza y posicionarse en la mente del cliente, y abarca la identidad visual de la marca, la experiencia que entrega y la percepción que las personas tienen de ella [40]. Este trabajo suele registrarse en un manual de marca (brand book), un documento que reúne los elementos que definen la marca, como su propósito, misión, valores, personalidad, logo, paleta cromática, tipografías y tono de comunicación, para que se utilicen de forma consistente en todos los canales.

Para orientar la marca hacia un cliente concreto, el material de la asignatura considera tres herramientas. El perfil de cliente ideal (buyer persona) describe al cliente objetivo

[9] Para efectos de este trabajo, se toma como Lean Canvas el formato de plantillas (templates) del 3IE, que se presenta en los anexos (ver Figura 17 a Figura 21).

respondiendo preguntas como quién es, qué necesita, qué problema tiene, qué redes utiliza, cómo busca información y qué le impide comprar. El posicionamiento define qué espacio quiere ocupar la marca en la mente de ese cliente y si ese espacio es relevante y defendible frente a la competencia. Finalmente, el mapa del recorrido del cliente (Customer Journey Map) es una representación visual de todas las etapas, interacciones y experiencias que vive el cliente desde que descubre una necesidad hasta que se convierte en promotor de la marca, habitualmente organizadas en descubrimiento, consideración, decisión, experiencia y fidelización. A diferencia de un diagrama de procesos interno, describe la experiencia desde el punto de vista del cliente [40].

Estas herramientas son las que estructuran el trabajo de emprendimiento que se propone en el capítulo siguiente. El Lean Canvas y el Business Model Canvas se trabajan durante el primer semestre, mientras que el manual de marca, el perfil de cliente ideal y el posicionamiento forman parte del Plan de Marketing del segundo semestre.

2.4. IA COMO HERRAMIENTA DE DESARROLLO

El desarrollo de software asistido por inteligencia artificial se refiere al uso de modelos de lenguaje de gran escala integrados en el flujo de trabajo de programación, ya sea dentro del editor de código, como asistente conversacional o como un agente capaz de ejecutar tareas de forma autónoma. Según cuánta autonomía se le entrega al modelo, se pueden distinguir tres niveles de uso. El primero es el autocompletado asistido, popularizado por herramientas como GitHub Copilot, que sugiere líneas o bloques de código a partir del contexto inmediato y deja al desarrollador el control total sobre cada decisión. El segundo es el llamado vibe coding, en el que el desarrollador conversa con el modelo y le da instrucciones en lenguaje natural, guiándolo y manteniendo el control del proceso, lo que resulta útil para prototipar y explorar ideas. El tercero es el agentic coding, en el que un agente de IA planifica, ejecuta, prueba e itera sobre tareas de desarrollo completas con mínima supervisión humana, y entrega como resultado cambios completos en el código y no solo sugerencias [41]. La diferencia entre estos niveles no es solo de nombre. El grado de autonomía que un equipo le cede al modelo determina en gran medida el impacto que su uso tiene sobre el proceso de desarrollo.

2.4.1. IMPACTO DE LA IA EN EL CICLO DE DESARROLLO

Más allá de la velocidad con que se genera código, la incorporación de asistentes de IA modifica la naturaleza misma del trabajo de desarrollo. Un estudio longitudinal de Vella y Blincoe [42] reporta que un 84% de los desarrolladores encuestados percibió mejoras de productividad y un 82% señaló dedicar menos tiempo a escribir código, pero que, en paralelo, la experiencia de desarrollo se deterioró de manera sostenida entre ambas mediciones del estudio, medida en términos de estado de flujo y carga cognitiva. Los autores

caracterizan este cambio como un desplazamiento desde actividades de creación hacia lo que denominan «supervisory engineering work»: la dirección, evaluación y corrección del resultado producido por la IA, en lugar de su escritura directa. Este desplazamiento tiene implicancias directas sobre los roles de un equipo de desarrollo, ya que tareas que antes se distribuían según quién implementaba cada componente pasan a depender de quién es capaz de evaluar críticamente si lo generado por la IA es correcto, se integra con el resto del sistema y respeta las decisiones de diseño acordadas por el equipo, lo cual es considerado una habilidad distinta a la de programar y que no todos los integrantes de un equipo desarrollan al mismo ritmo.

Este mismo desplazamiento ayuda a explicar por qué la asistencia de IA tiende a debilitar, en lugar de fortalecer, las prácticas de documentación técnica de un proyecto. Fowler [43] describe este fenómeno como una «trampa de la implementación». Como un asistente de IA puede producir código funcional casi de inmediato, desaparece el punto de control en que un equipo se detiene a discutir explícitamente su arquitectura antes de escribirla, de modo que decisiones como la estructura del modelo de datos o los límites entre componentes del sistema quedan implícitas dentro del código generado en lugar de ser documentadas y acordadas de forma explícita. En proyectos de la Feria de Software esto se traduce en un riesgo concreto. Sin un punto de control que obligue a explicitar el modelo de base de datos o el diagrama de arquitectura antes de avanzar en la implementación, ambos documentos tienden a omitirse o a quedar desactualizados respecto del código que efectivamente terminó generando la IA, comprometiendo tanto la comprensión del sistema por parte del propio equipo como la calidad de los entregables de documentación que se evalúan al cierre del proyecto.

2.5. DIVERSIFICACIÓN DE PROYECTOS

Antes de continuar, conviene precisar la relación entre tres términos que suelen usarse como sinónimos, pero que no lo son: Inteligencia Artificial, Machine Learning y Ciencia de Datos. La Ciencia de Datos es el campo más amplio de los tres. Combina matemáticas y estadística, programación especializada y conocimiento del dominio del negocio para descubrir más conocimiento y generar valor a partir de los datos de una organización, e incorpora tanto la Inteligencia Artificial como el Machine Learning como parte de sus herramientas, no como sinónimos de la disciplina completa [44]. El Machine Learning, a su vez, es un subconjunto específico de la Inteligencia Artificial: un conjunto de algoritmos capaces de aprender los patrones presentes en datos de entrenamiento para, a partir de ellos, realizar inferencias sobre datos nuevos, de modo que todo Machine Learning es Inteligencia Artificial, pero no toda Inteligencia Artificial es Machine Learning [45]. Esta distinción importa para la propuesta porque el criterio de clasificación de naturaleza de proyecto que se propone en el capítulo siguiente no depende de si un proyecto usa datos, ni de si se enmarca de forma general dentro de la Ciencia de Datos o la Inteligencia Artificial, sino específicamente de si

su desarrollo requiere entrenar y evaluar un modelo de Machine Learning, ya que es este subconjunto el que exige un ciclo de vida propio.

Precisamente porque el Machine Learning exige un ciclo de vida propio, conviene situar ambos tipos de proyecto dentro de los ciclos de vida que la industria ha adoptado para gestionarlos, antes de abordar por qué uno no se ajusta a los mismos lineamientos de desarrollo y evaluación que el otro. En el desarrollo de software convencional, la práctica actual converge hacia el modelo DevOps, que integra las labores de desarrollo (Dev) y de operaciones (Ops) en un ciclo continuo, descrito habitualmente como un «bucle infinito» de ocho fases: planificar, codificar, construir, probar, liberar, desplegar, operar y monitorear. Su objetivo es acortar el tiempo entre una necesidad identificada y su solución entregada en producción, mediante integración y entrega continuas (CI/CD) [46]. Este ciclo asume que el artefacto que se construye y despliega es determinista, es decir, que ante la misma entrada produce el mismo resultado, lo que permite pruebas y liberaciones incrementales relativamente predecibles.

Un proyecto que entrena su propio modelo de Machine Learning extiende y complejiza este ciclo. La disciplina de MLOps (Machine Learning Operations) surge precisamente para responder a que, además del código, un sistema de ML depende de los datos y del modelo entrenado, cada uno versionable y sujeto a degradación en el tiempo. MLOps agrega al ciclo DevOps etapas propias de este dominio como la extracción y validación de datos, el entrenamiento y validación del modelo, y el monitoreo continuo de su desempeño en producción, ya que un modelo que funciona correctamente al desplegarse puede perder precisión con el paso del tiempo a medida que los datos del mundo real se alejan de los datos con que fue entrenado [47].

A nivel de gestión del proyecto ocurre algo parecido. Un proyecto de Machine Learning no encaja con los mismos lineamientos de desarrollo y evaluación que uno de software tradicional. Scrum asume que cada sprint entrega un incremento de producto funcional y evaluable, pero en un proyecto de ML, gran parte del esfuerzo inicial se dedica a explorar si los datos disponibles realmente permiten resolver el problema planteado. Esta exploración es incierta y no siempre termina en algo entregable o visible al cierre del sprint, por lo que se ajusta mejor a una etapa de exploración acotada en el tiempo, es decir, se parece más a una especie de Spike que a un ciclo de desarrollo convencional [13], [48].

2.5.1. MODELOS DE PROCESO PARA PROYECTOS DE DATOS

Así como el desarrollo de software cuenta con los marcos ágiles descritos anteriormente, los proyectos centrados en datos cuentan con sus propios modelos de proceso, que ordenan el trabajo técnico desde la obtención de los datos hasta el uso del modelo. Para elegir en cuál apoyar la propuesta, se revisaron cuatro de los modelos más difundidos.

El proceso KDD (Knowledge Discovery in Databases) fue el primero en formalizar este trabajo, como una secuencia de pasos que va desde la selección de los datos hasta la interpretación de los patrones encontrados [49]. SEMMA (Sample, Explore, Modify, Model, Assess), propuesto por SAS, organiza el trabajo en cinco pasos centrados en el tratamiento de los datos y el modelado [50]. OSEMN (Obtain, Scrub, Explore, Model, iNterpret), planteado por Mason y Wiggins en 2010, es una taxonomía más reciente y popular en ciencia de datos [51]. Por último, CRISP-DM (Cross-Industry Standard Process for Data Mining) define seis fases que van desde la comprensión del negocio hasta el despliegue [3].

Los cuatro modelos se compararon según tres criterios relevantes para la Feria de Software (ver Tabla 1). El primero es si dependen de una herramienta específica, ya que los equipos eligen libremente sus tecnologías. El segundo es si incluyen una fase de comprensión del negocio y una de despliegue, porque en la Feria el modelo no es un fin en sí mismo, sino parte de un producto que debe responder a un problema y llegar a manos de un usuario. El tercero es su adopción en la práctica, medida según una encuesta de KDnuggets a profesionales del área [52].

Modelo	**¿Depende de una herramienta?**	**¿Incluye comprensión del negocio y despliegue?**	**Uso declarado (KDnuggets 2014)**
KDD	No	No, en su versión resumida [53].	7,5%
SEMMA	Sí, está ligado a SAS Enterprise Miner [53].	No.	8,5%
OSEMN	No	No. Parte en la obtención de los datos y termina en su interpretación.	No medido
CRISP-DM	No	Sí.	43%

Tabla 1: Comparación de modelos de proceso para proyectos de datos. Fuente: Elaboración propia.

CRISP-DM es el único de los cuatro que cumple los tres criterios. Además, la literatura lo describe como la metodología más utilizada para proyectos de minería de datos [53], y fue el modelo más declarado por los profesionales en la encuesta de KDnuggets [52]. Una revisión posterior concluye que sigue siendo adecuado para proyectos orientados a un objetivo, mientras que los proyectos puramente exploratorios requieren un modelo más flexible [54]. Los proyectos de la Feria pertenecen al primer tipo, ya que parten de un problema concreto y de una propuesta de valor definida. Su principal limitación es que no cubre las actividades de gestión del proyecto [53], y es justamente esa limitación la que justifica combinarlo con Scrum en lugar de reemplazarlo, como se detalla más adelante.

2.5.2. CICLO DE VIDA DE UN PROYECTO DE MACHINE LEARNING

CRISP-DM, el modelo seleccionado en la sección anterior, es un modelo de proceso independiente de la industria y de la herramienta utilizada, compuesto por seis fases que se ejecutan de forma cíclica y no estrictamente secuencial (ver Figura 5) [3]:

1. Comprensión del negocio, en la que se traduce el problema a resolver en objetivos y requisitos claros desde la perspectiva del proyecto.
2. Comprensión de los datos, que consiste en recolectar y explorar los datos disponibles para identificar su calidad y detectar problemas tempranos.
3. Preparación de los datos, en la que se limpian, transforman y estructuran los datos para dejarlos listos para el modelado.
4. Modelado, en la que se seleccionan y aplican distintas técnicas de aprendizaje automático sobre los datos preparados.
5. Evaluación, en la que se valida si el modelo resultante responde efectivamente a los objetivos definidos al inicio y no solo a una métrica técnica.
6. Despliegue, en la que el modelo se organiza y presenta de forma que el conocimiento obtenido pueda usarse en la práctica.

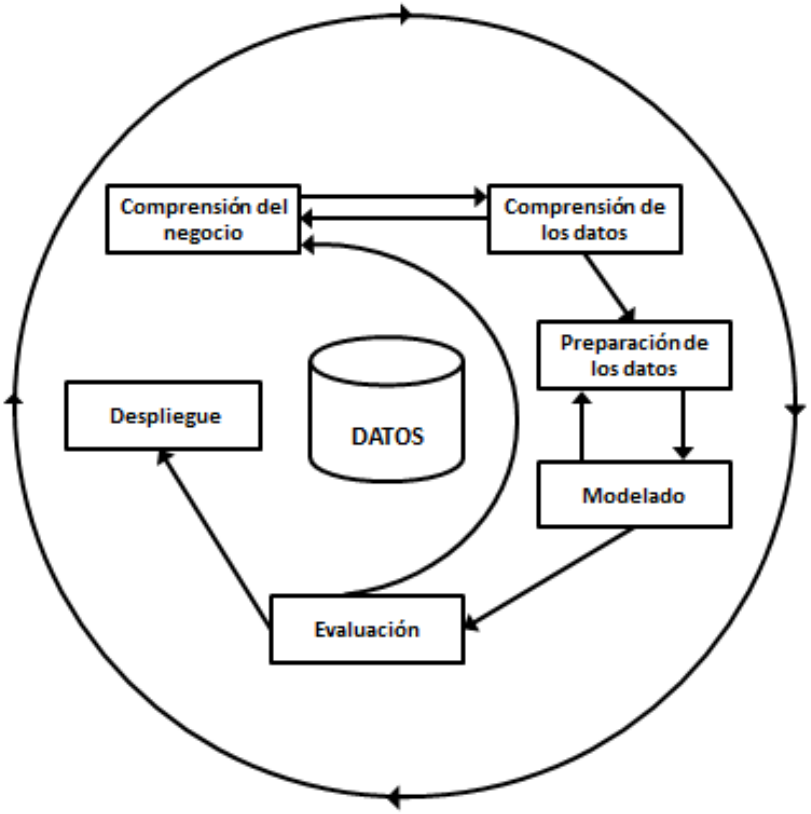


Figura 5: Ciclo de vida CRISP-DM. Fuente: Adaptado y traducido de Wirth y Hipp [3].

Como muestra la Figura 5, el flujo parte en la comprensión del negocio y avanza hacia la comprensión de los datos, con una flecha en ambos sentidos entre ellas, porque lo que se

descubre en los datos puede obligar a replantear los objetivos. Luego pasa a la preparación de los datos y al modelado, que también se retroalimentan, ya que cada técnica de modelado puede exigir que los datos se preparen de otra forma. Desde el modelado se llega a la evaluación, que puede devolver el proceso a la comprensión del negocio si el modelo no responde a los objetivos, o dar paso al despliegue. Los datos se ubican al centro porque todas las fases trabajan sobre ellos, y el círculo exterior indica que el ciclo completo puede repetirse cuando el despliegue abre nuevas preguntas.

A diferencia del ciclo de vida del software tradicional, CRISP-DM contempla explícitamente el retorno a fases anteriores cuando los resultados de una etapa posterior (por ejemplo, una evaluación insatisfactoria del modelo) revelan que los datos o la comprensión del problema deben revisarse, lo que hace que su avance sea naturalmente menos lineal que el de un incremento de sprint [3].

Sobre esta base, extensiones más recientes como CRISP-ML(Q) incorporan de forma explícita el aseguramiento de la calidad como una preocupación transversal a todas las fases, agregando actividades como la definición de requisitos de calidad del modelo desde la etapa de comprensión del negocio y el monitoreo de su desempeño una vez desplegado, cerrando así la brecha entre un modelo que funciona en un ambiente de prueba y uno que se mantiene confiable en producción [55]. Esta preocupación por sostener la calidad del modelo en el tiempo es consistente con lo señalado por MLOps para el ciclo de vida general del software con componentes de IA.

En la práctica, la integración de CRISP-DM con marcos ágiles como Scrum no es excluyente: existen propuestas que combinan ambos, usando las fases de CRISP-DM para estructurar el contenido técnico del trabajo (qué se hace en cada etapa del proyecto de datos) y Scrum para organizar la cadencia de equipo (cómo se planifica, ejecuta y revisa ese trabajo en el tiempo), de modo que cada sprint se enfoque en avanzar una o más fases de CRISP-DM en lugar de asumir que producirá un incremento de producto terminado [48]. Esta combinación resulta especialmente relevante para la propuesta, en la medida en que sugiere que la adecuación del flujo de procesos para proyectos con modelo propio no requiere abandonar Scrum, sino ajustar qué se entiende por «incremento» y por «hito» dentro de cada sprint cuando el producto central es un modelo y no una funcionalidad de software tradicional.

2.6. DATOS PERSONALES EN PROYECTOS ACADÉMICOS

Un factor adicional que la propuesta vigente no contempla, porque tampoco fue relevante en 2019, es el nuevo marco legal chileno de protección de datos personales. La Ley 21.719, publicada en el Diario Oficial el 13 de diciembre de 2024 y que entrará en vigencia plena el 1 de diciembre de 2026, establece un principio de licitud y lealtad en el tratamiento de datos personales, un consentimiento informado y específico para cada finalidad de uso, los derechos ARCO+P (acceso, rectificación, cancelación, oposición y portabilidad) para los

titulares de los datos y la creación de una Agencia de Protección de Datos Personales encargada de fiscalizar su cumplimiento [56].

Este cambio es directamente relevante para los proyectos de la Feria de Software, que en un número creciente de casos manejan datos personales reales: proyectos en la categoría salud que procesan información de pacientes, proyectos educativos que registran datos de estudiantes, o soluciones que simplemente requieren un sistema de usuarios con datos de contacto. A diferencia de una empresa, los equipos que desarrollan estos proyectos no cuentan con un área legal ni con experiencia previa en cumplimiento normativo, y como se discutió en la sección *Impacto de la IA en el ciclo de desarrollo* sobre el impacto de la IA en la documentación, es habitual que decisiones de diseño (incluyendo qué datos se almacenan y con qué nivel de protección) queden implícitas en el código generado en lugar de ser evaluadas y documentadas explícitamente. A partir de diciembre de 2026, esto deja de ser un riesgo teórico para convertirse en una obligación legal concreta que cualquier adecuación del flujo de procesos de la Feria de Software debiera al menos reconocer. Por ello, la propuesta incorpora dentro de los sprints una actividad dedicada a comprobar la seguridad de los datos que manejan los proyectos, aunque definir cómo se realiza esa comprobación y los controles de cumplimiento específicos frente a la ley excede su alcance y queda planteado como trabajo futuro.

2.7. ADAPTACIÓN DE MARCOS ÁGILES

La propuesta no busca reemplazar el marco vigente de la Feria de Software ni introducir una metodología distinta, sino adecuarlo. Antes de continuar, conviene precisar qué se entiende por adecuar en este contexto: según la Real Academia Española, adecuar significa «adaptar algo a las necesidades o condiciones de una persona o de una cosa» [57]. A diferencia de rediseñar o reemplazar, adecuar presupone conservar la estructura y el propósito de aquello que se adapta, ajustando solo aquellos puntos que han dejado de responder a las condiciones actuales.

Esta lógica no es ajena al marco vigente. El propio Calvo [1], al proponer en 2019 un framework «basado en Scrum» y no en Scrum en su forma pura, ya reconocía que una adaptación deliberada y justificada del marco ágil era preferible a una adopción literal, dadas las restricciones del entorno académico. La literatura sobre metodologías ágiles en educación confirma que esta es la norma, no la excepción: Masood, Hoda y Blincoe [29] documentan que los equipos universitarios ajustan sistemáticamente prácticas puntuales de Scrum (como la frecuencia del Daily Scrum) sin abandonar el marco completo, en respuesta a fricciones estructurales específicas y no por conveniencia arbitraria.

Este trabajo se inscribe en esa misma tradición de adaptación. No parte de la premisa de que el marco definido en 2019 esté mal diseñado, sino de que las condiciones bajo las cuales fue diseñado han cambiado en al menos tres aspectos concretos, las exigencias curriculares en innovación y emprendimiento, la diversificación de los proyectos hacia

Machine Learning, y el rol de la inteligencia artificial como herramienta de desarrollo, identificados y fundamentados a lo largo de este capítulo. Adecuar el flujo de procesos significa, en este sentido, identificar con precisión en qué etapas concretas del marco de Calvo [1] estas tres condiciones generan un desajuste y ajustar únicamente esos puntos, preservando la estructura, los roles y el propósito formativo del resto del proceso.

2.8. TRABAJOS RELACIONADOS

El marco vigente de la Feria de Software no surgió de una sola vez, sino como resultado de una serie de trabajos de título sucesivos que fueron diagnosticando y proponiendo soluciones parciales al mismo problema de fondo: cómo gestionar proyectos de emprendimiento temprano dentro de un entorno académico con restricciones propias. El primero en identificar formalmente este problema fue Chehade, en 2013, quien, según recoge Moya [58], diagnosticó que el entonces Taller de Desarrollo de Software se encontraba en una etapa de madurez incipiente en materia de gestión de proyectos: si bien se reconocía teóricamente su importancia, no existía un uso efectivo de prácticas de gestión por parte de profesores ni ayudantes, y el seguimiento se limitaba a unas pocas evaluaciones de hitos.

Tomando ese diagnóstico como base, en 2017 Moya [58] implementó un piloto de Oficina de Gestión de Proyectos (PMO) de tipo ágil, basada en Scrum, enfocado específicamente en resolver el problema de seguimiento y control durante la etapa de desarrollo del proyecto, mediante reuniones estandarizadas, reportes de estado e indicadores de adherencia metodológica. A diferencia de lo que vendría después, su propuesta no definía el proceso completo del taller ni sus roles, sino que se limitaba a monitorear su ejecución. Aun así, sus resultados ya advertían que el control de los requerimientos y del alcance del proyecto podía perderse tanto durante el desarrollo como durante la documentación, un riesgo que, como se detalla más adelante en este capítulo, sigue presente en la operación actual de la Feria de Software [58].

Dos años más tarde, en 2019, Calvo [1] formalizó el marco que rige hasta hoy: una estructura basada en Scrum que organiza el ciclo completo de ambas asignaturas en diez etapas agrupadas en cuatro categorías, con roles explícitos para profesores, ayudantes y equipos de estudiantes. A diferencia de la propuesta de Moya, centrada únicamente en el seguimiento de una etapa, la de Calvo definió el proceso de principio a fin, y es sobre esta base que se apoya el diagnóstico de este trabajo.

El trabajo más reciente en esta línea, de 2025, es el de Cayo [2], quien se concentró en fortalecer específicamente el componente de innovación dentro del marco de Calvo, incorporando un conjunto de herramientas de Design Thinking en las primeras semanas de GPI para sistematizar una etapa que hasta entonces solo se evaluaba como un ítem más dentro de la rúbrica de postulación.

Leídos en conjunto, estos cuatro trabajos muestran una progresión: de un diagnóstico general (Chehade), a una solución parcial de seguimiento (Moya), a un marco de proceso completo (Calvo), a un refuerzo específico de una de sus etapas (Cayo). Este trabajo se inserta en esa misma línea, pero con un alcance distinto: no busca diagnosticar el problema desde cero, ni reemplazar el marco vigente, ni fortalecer una sola de sus etapas, sino adecuar el flujo de procesos completo definido por Calvo en 2019 a tres condiciones que ninguno de estos trabajos previos contempló, porque no existían o no eran relevantes al momento de su publicación: las mayores exigencias curriculares en innovación y emprendimiento del nuevo plan de estudios, la diversificación de los proyectos hacia soluciones centradas en Machine Learning, y el rol que hoy cumple la inteligencia artificial como herramienta dentro del propio proceso de desarrollo.

2.9. SÍNTESIS DE LA PROBLEMÁTICA

A lo largo de este capítulo se revisaron los fundamentos sobre los que se apoya el marco vigente de la Feria de Software, junto con las condiciones que hoy rodean a sus proyectos y que no existían, o no eran relevantes, cuando ese marco se definió en 2019. Esta sección reúne esos elementos y los contrasta con lo que efectivamente ocurre en GPI y TDPI, a partir de tres fuentes: lo que plantea la literatura revisada, la experiencia de la autora como estudiante y como ayudante de ambas asignaturas, como contexto complementario, y la evolución de los proyectos presentados en la Feria de Software entre 2018 y 2025.

2.9.1. LO QUE PLANTEA LA LITERATURA

La literatura sobre cursos Capstone coincide en que llevar Scrum a un contexto universitario genera tres fricciones estructurales: la dedicación parcial de los estudiantes, que compiten con la carga del resto de sus asignaturas, un calendario fijo, que no depende del avance real del producto y una lógica de evaluación individual que no siempre se alinea con el objetivo de maximizar el valor del producto [6], [7], [29].

A estas fricciones se suman tres condiciones propias del contexto actual:

- **Proyectos que entrenan y evalúan un modelo propio:** Su ciclo de vida es exploratorio y no lineal, y contempla volver a fases anteriores cuando la evaluación del modelo no da los resultados esperados [3]. Eso choca con la idea de que cada sprint entregue un incremento funcional.
- **La IA como herramienta de desarrollo:** Cambia la forma de trabajo de los equipos desde escribir código hacia supervisar y corregir lo que genera la IA [42], y tiende a eliminar el momento en que un equipo discute y documenta su arquitectura antes de implementarla [43].

- **Más exigencia en innovación y emprendimiento:** Enseñar a innovar no garantiza que los estudiantes aprendan a evaluar si su idea es viable como negocio [36]. Esa brecha también existe en la Feria de Software, el trabajo de Cayo [2] sistematizó la etapa de innovación, pero no cubrió la validación de la idea como oportunidad de negocio, ni cómo se desarrolla el proceso de emprendimiento.

2.9.2. LA EXPERIENCIA COMO ESTUDIANTE Y COMO AYUDANTE

Lo planteado por la literatura puede complementarse con la experiencia de la autora desde dos posiciones distintas dentro de la Feria de Software, como estudiante que cursó GPI y TDPI y, posteriormente, como ayudante de ambas asignaturas. Esta experiencia no permite generalizar por sí sola, pero aporta contexto y conocimiento del curso para interpretar lo que plantea la literatura y la revisión de proyectos.

Como estudiante, la autora vivió de primera mano dos situaciones que se relacionan directamente con esta propuesta. La primera tiene que ver con la documentación técnica. Su equipo esbozó un modelo de base de datos al comienzo del proyecto, pero nunca lo actualizó a medida que el alcance cambiaba, porque ninguna entrega del curso lo exigía. La segunda tiene que ver con el uso de herramientas de IA, que hacia el final del semestre y bajo presión de tiempo permitió avanzar más rápido, pero dejó funciones o bloques de código duplicados que el equipo no alcanzó a depurar.

Como ayudante, la autora ocupa el rol de Jefe de proyectos que define Calvo [1] para esta figura, aunque en la práctica se ejerce más como un rol de gestión de hitos y consejería que de supervisión técnica. La mayoría de las consultas que recibe de los equipos se orientan a mantenerse al día con las entregas del curso, y rara vez se refieren a código o arquitectura del proyecto, lo cual tiene sentido con el rol que se posee, dado que la revisión técnica no forma parte de sus funciones. Respecto a los proyectos con un componente de Machine Learning, ha observado distintos equipos que, presionados por la entrega del PMV, pausan el ajuste o la validación del modelo para priorizar las historias de usuario, entregando avance visible por sobre calidad técnica. Esta experiencia es consistente con lo que plantea la literatura sobre el ciclo de vida de estos proyectos, y sugiere que la etapa de Prototipo, pensada originalmente para software tradicional, ya no calza de la misma forma para ellos. En su opinión, esta etapa debería estructurarse de forma distinta. En cuanto al emprendimiento, ha observado equipos que no cuentan con una base para presentar un modelo de negocios a su cliente. Dudan de cuánto vale su idea y su proyecto, y no saben cómo llegar a un acuerdo con el cliente, en parte porque el modelo de negocios se les pide recién después del segundo sprint, cuando el proyecto ya está avanzado. Dejar este modelo establecido desde el inicio les daría una base para valorar su proyecto y negociar con el cliente, y también para continuar con su startup después de la Feria de Software, si así lo deciden. Finalmente, ha observado que la omisión del modelo de base de datos se repite en otros equipos, por lo que no parece ser

un caso aislado de su propio equipo. Todos saben que corresponde hacerlo, pero al no ser parte de ninguna entrega formal, se posterga en favor de otras prioridades.

Estas dos miradas apuntan en una misma dirección. Los problemas observados son consistentes con lo que plantea la literatura y parecen sostenerse porque ningún hito del flujo actual exige explícitamente resolverlos, ni la documentación técnica del proyecto, ni una diferenciación real entre lo que necesita un proyecto de software tradicional y uno con un componente de Machine Learning, ni un proceso de emprendimiento que esté estandarizado.

2.9.3. EVOLUCIÓN DE LOS PROYECTOS DE LA FERIA DE SOFTWARE

Para dimensionar cuánto ha cambiado el tipo de proyecto que llega a la Feria de Software desde que se definió el marco vigente, se revisaron los proyectos publicados en el sitio oficial de la Feria para cada edición entre 2018 y 2025 [11]. Se partió en 2018 porque es el último año del período que diagnosticó Calvo [1] para construir su propuesta.

Cada proyecto se clasificó en una de dos categorías: software tradicional, o proyecto con algún componente de inteligencia artificial o Machine Learning (modelos predictivos, visión computacional, procesamiento de lenguaje natural, asistentes basados en modelos de lenguaje, entre otros). Un proyecto se consideró dentro de la segunda categoría cuando su página lo declaraba, ya fuera en la descripción o en las etiquetas asignadas. Entre 2019 y 2023 la Feria contó además con una categoría oficial de «Inteligencia Artificial» o «Inteligencia Artificial y Ciencia de Datos», en la que cada equipo inscribía su proyecto. Entre 2019 y 2022 las páginas de los proyectos solo contienen una frase descriptiva, por lo que en varios casos la descripción no menciona una función de IA. Sin embargo, al haber sido inscritos por sus propios equipos en una categoría asociada a la inteligencia artificial, estos proyectos también se consideraron dentro de la segunda categoría y se incluyeron en el porcentaje (ver Tabla 2).

Edición	Total	Con IA/ML	Software tradicional	% con IA/ML
2018	16	3	13	19%
2019	16	9	7	56%
2020	20	10	10	50%
2021	22	11	11	50%
2022	25	13	12	52%
2023	28	18	10	64%
2024	31	23	8	74%
2025	35	23	12	66%

Tabla 2: Proyectos de la Feria de Software según componente de IA/ML, 2018–2025. Fuente: Elaboración propia a partir del sitio oficial de la Feria de Software.

En 2018, tres de los dieciséis proyectos presentados incorporaban algún componente de inteligencia artificial. Desde 2019, en cambio, al menos la mitad de los proyectos de cada edición lo hace, y entre 2023 y 2025 la proporción sube a 64 de 94, es decir, cerca de dos de cada tres. La propia Feria reflejó este cambio en su forma de organizar los proyectos: entre 2019 y 2023 existió una categoría dedicada exclusivamente a la inteligencia artificial, y desde 2024 esa categoría desaparece porque la IA pasa a estar presente de forma transversal en todas las demás (industria, educación y salud). El marco vigente fue diseñado a partir de una realidad en la que este tipo de proyecto era la excepción, hoy es la mayoría.

Esta clasificación tiene una limitación. Se basa en lo que los equipos declaran sobre su proyecto y no en una revisión de su implementación, y algunos equipos etiquetan su proyecto con «Inteligencia Artificial» sin describir qué función cumple. Aun así, la tendencia es lo suficientemente marcada como para respaldar que la diversificación de proyectos descrita en el capítulo 1 no es un caso aislado, sino la realidad actual de la Feria. Cabe precisar que esta categoría es más amplia que el criterio con que la propuesta clasifica la naturaleza de un proyecto en el capítulo 3. Un proyecto que integra un modelo de lenguaje u otro modelo ya entrenado mediante un servicio externo cuenta aquí como proyecto con componente de IA, pero no requiere entrenar ni evaluar un modelo propio, por lo que en la propuesta sigue el camino sin modelo propio. La tabla, por lo tanto, no indica cuántos proyectos seguirían el camino de Machine Learning, sino cuánto ha crecido la presencia de la IA en los proyectos de la Feria.

2.9.4. DESAJUSTES IDENTIFICADOS

Al cruzar las tres fuentes anteriores con las etapas del marco de Calvo [1] (ver Figura 2), los desajustes se concentran en los siguientes puntos:

1. **Realización del Prototipo:** Los proyectos con modelo propio deben mostrar un avance visible en una etapa donde su trabajo real es explorar si los datos permiten resolver el problema, lo que lleva a postergar el modelo en favor de la interfaz.
2. **Desarrollo del Producto (Sprints 1 a 3):** La definición de incremento asume un producto funcional al cierre de cada sprint, sin espacio para el avance no lineal de un modelo ni un punto que permita ajustar el alcance si el modelo no alcanza el desempeño esperado.
3. **Planificación tentativa y Sprints:** Ninguna entrega exige definir ni mantener actualizada la documentación, una omisión que el uso de herramientas de IA en el desarrollo hace más probable.
4. **Emprendimiento a lo largo de GPI y TDPI:** La innovación se trabaja en la etapa temprana de GPI, pero el emprendimiento no tiene hoy un proceso definido a lo largo de ambas asignaturas. Más allá del Business Model Canvas y el modelo de negocio que se piden después del segundo sprint, no existe una secuencia de actividades y entregables que acompañe al equipo en la construcción de su proyecto como posible emprendimiento. Por eso, si al terminar la Feria de Software el equipo quiere continuar con el desarrollo

de su startup, no cuenta con una base sólida y ordenada para presentarla ante un posible cliente, inversionista o comprador.

5. **Datos personales e inteligencia artificial:** Los proyectos que manejan datos de pacientes, estudiantes o usuarios no tienen ningún punto de control frente a las exigencias de la Ley 21.719 [56]. Este riesgo crece con la incorporación de IA. Como muestra la Tabla 2, hoy cerca de dos de cada tres proyectos tienen un componente de IA, y varios de ellos procesan datos sensibles para entrenar o alimentar sus modelos [11]. A esto se suma que el uso de herramientas de IA durante el desarrollo puede llevar a que esos datos se compartan con servicios externos, sin que el equipo evalúe qué información sale del sistema ni con qué resguardo.

Esta síntesis da cuenta del diagnóstico planteado en los objetivos específicos 1 y 2, y los desajustes identificados son los que aborda la propuesta de adecuación que se desarrolla en el capítulo siguiente. El objetivo específico 3 se aborda en la sección *Metodologías y herramientas ágiles en la industria*, donde se contrasta el marco vigente con los enfoques ágiles actuales de la industria.

CAPÍTULO 3
PROPUESTA DE SOLUCIÓN

3.1. ENFOQUE DE LA PROPUESTA

Esta propuesta no busca reemplazar el marco de la Feria de Software definido por Calvo [1], sino adecuarlo, conservando su estructura, roles y propósito formativo, y ajustando únicamente los puntos donde hoy se genera un desajuste real entre el proceso definido y la realidad de los proyectos que lo atraviesan. Como se fundamentó en el marco conceptual, hoy en día son tres las condiciones que más aportan a este desajuste. La primera son las mayores exigencias curriculares en innovación y emprendimiento del nuevo plan de estudios, la segunda es la diversificación de los proyectos hacia soluciones centradas en modelos de Machine Learning, y la tercera es el papel que actualmente cumple la inteligencia artificial como herramienta dentro del propio proceso de desarrollo. Este capítulo documenta, para cada una de estas tres condiciones, en qué etapas concretas del flujo se concentra el desajuste correspondiente y qué ajuste puntual se propone para resolverlo, sin alterar aquello que sigue funcionando.

Como se definió en el marco conceptual, adecuar significa adaptar algo a las necesidades o condiciones actuales, conservando la estructura y el propósito de aquello que se adapta (ver *Adaptación de marcos ágiles*). Es la misma lógica que siguió Calvo [1] al definir su marco como «basado en Scrum» y no como Scrum en su forma pura, y esta propuesta se inscribe en esa tradición de adaptación, no de reemplazo.

Para documentar esta adecuación, el flujo anual se organiza en procesos por semestre. El primer semestre se compone de tres procesos consecutivos y el segundo de los tres sprints, el evento de Feria de Software y el cierre de proyectos, con el proceso de Emprendimiento en paralelo. Cada proceso se describe con un mismo formato, que incluye su objetivo, su flujo, sus entradas y salidas, sus reglas de negocio y una descripción. Los entregables que se modifican o se crean dentro de cada proceso se describen con su objetivo, sus entradas y salidas, y una plantilla que propone su estructura. Este formato permite distinguir con claridad qué se conserva del marco vigente y qué cambia en cada etapa.

Las ceremonias de Scrum (Sprint Review y Sprint Retrospective), los roles que el marco vigente asigna a profesores, ayudantes y equipos, y el calendario general del año se mantienen, sin importar la naturaleza del proyecto. Lo que se adapta es, principalmente, el contenido del backlog, los formularios de entrega y la composición de los documentos formales. Cuando el ajuste de contenido no basta para resolver un desajuste, la propuesta agrega elementos puntuales al flujo, como una etapa de clasificación de la naturaleza del proyecto después de la Licitación, actividades paralelas dentro de los sprints (Actualización de documentación y Comprobación de seguridad de los datos) y un proceso de Emprendimiento con sus propios entregables. Cada uno de estos elementos responde

a un desajuste identificado en la síntesis de la problemática del capítulo anterior y se justifica en la sección correspondiente. Esto resuelve la pregunta de qué significa adecuar el flujo de proceso. No se trata de definir procesos paralelos para cada condición nueva, uno para software tradicional y otro para Machine Learning, uno para innovación y otro para emprendimiento, sino de identificar con precisión qué cambia en qué etapa puntual, conservando la misma mecánica que ya conocen y ejecutan profesores, ayudantes y estudiantes.

Además de estos elementos, la propuesta ajusta el criterio de una instancia que ya existe y que es común a ambas naturalezas de proyecto. El prototipo inicial que hoy se presenta durante el primer semestre pasa a llamarse PMV y debe demostrar la propuesta de valor del proyecto, en lugar de mostrar solo historias de alto valor pero de menor dificultad. De esta forma, la presentación funciona como una mitigación de riesgos antes de planificar el segundo semestre, formalizando una práctica que el curso ya venía aplicando de manera informal.

Para la condición curricular de innovación y emprendimiento, el componente de innovación no requiere ajustes en su contenido, ya que, como se detalla en la sección *Propuesta de innovación y definición del problema*, fue sistematizado por Cayo [2] dentro de las primeras etapas del primer semestre. Lo que esta propuesta formaliza es el registro de esas etapas mediante las plantillas de Lean Canvas del 3IE. El emprendimiento, en cambio, sí se formaliza como un proceso explícito. Al cierre del primer semestre el equipo consolida su modelo de negocios, con su Business Model Canvas completo y una meta de clientes, y durante el segundo semestre, en paralelo a los sprints, diseña su marca, define un Plan de Marketing y lo sigue en tres hitos de seguimiento (Reportes de Marketing 1 a 3), sin ocupar ceremonias propias ni desplazar las ya existentes.

Para la condición de diversificación hacia proyectos con modelo propio, la adecuación se materializa mediante una etapa de clasificación de naturaleza de proyecto, situada después de la Licitación y antes del Sprint 0, que determina si el equipo seguirá el camino de los proyectos sin entrenamiento de modelo propio o el de los proyectos con entrenamiento y evaluación de modelo propio, que agrega al desarrollo de software las actividades del ciclo CRISP-DM. A partir de esa clasificación, el contenido del Sprint 0, del PMV, del Documento de Diseño y de los Sprints 1 y 2 se adapta según corresponda, sin que cambien las ceremonias ni los plazos del semestre. Al cierre del Sprint 2, junto a la decisión de continuidad que los profesores ya toman hoy, los proyectos con modelo propio que no cumplieron su criterio de éxito reducen el alcance de su objetivo de Machine Learning para el Sprint 3, en lugar de sacrificar el sprint completo. Ambos caminos convergen de vuelta en un único proceso a partir del Sprint 3, cuando se espera que las seis fases de CRISP-DM hayan sido abordadas al menos una vez.

Finalmente, para la condición del rol de la inteligencia artificial como herramienta de desarrollo, el ajuste no se concentra en una etapa propia, sino que atraviesa transversalmente el Documento de Diseño, que reemplaza al actual Plan de Proyecto, y la

actividad de Actualización de documentación que se repite en los tres sprints del segundo semestre. Como se detalla más adelante, ambos operan como el punto de control que evita que las decisiones de arquitectura queden implícitas dentro del código generado con asistencia de IA, en lugar de discutirse y registrarse explícitamente [43]. A esto se suma la Comprobación de seguridad de los datos, que desde el Sprint 2 revisa cómo el producto almacena y trata los datos sensibles, un riesgo que crece con la incorporación de IA en los proyectos [56].

Este ajuste responde también al impacto de la IA sobre los roles descrito en el marco conceptual. Los roles formales del marco vigente (profesores, ayudantes y equipo) no cambian, pero dentro del equipo el trabajo se desplaza desde escribir código hacia dirigir, revisar y corregir lo que genera la IA [42]. Por ello la propuesta no asigna esa revisión a un integrante en particular, sino que la convierte en una responsabilidad compartida. El Documento de Diseño fija las decisiones de arquitectura que todo el equipo debe conocer, y la Actualización de documentación obliga, en cada sprint, a contrastar lo que se generó con esas decisiones. El resto de este capítulo desarrolla estas condiciones a través de los procesos de cada semestre.

3.2. ESTRUCTURA PRIMER SEMESTRE

Durante el primer semestre el objetivo es que el equipo pase de identificar un problema a contar con un proyecto definido, con un alcance acotado y realista según su naturaleza y con un diseño aprobado para desarrollarlo durante el segundo semestre.

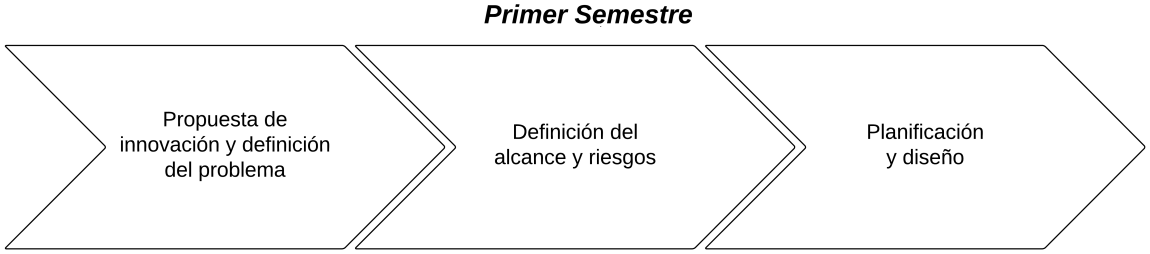


Figura 6: Procesos del primer semestre. Fuente: Elaboración propia.

El primer semestre se organiza en tres procesos que se ejecutan uno a continuación del otro (ver Figura 6). Cada proceso tiene su propio flujo, con su objetivo, entradas y salidas, y deja un resultado concreto que sirve de entrada al siguiente. Las secciones que siguen detallan cada uno de ellos.

- **Propuesta de innovación y definición del problema**: el equipo recorre las cuatro etapas de innovación definidas por Cayo [2], construye su Lean Canvas con el formato del 3IE, desarrolla su documento de propuesta y lo presenta en la Licitación. El proceso termina cuando los profesores aprueban el proyecto para su continuación.

- **Definición del alcance y riesgos**: profesores y ayudantes clasifican el proyecto según su naturaleza y, a partir de esa clasificación, el equipo define su Sprint 0 mediante historias de usuario o mediante un Spike, que culmina con la presentación del PMV. El proceso termina cuando el equipo demuestra su propuesta de valor.
- **Planificación y diseño**: el equipo consolida su Modelo de Negocios y desarrolla el Documento de Diseño, que los profesores retroalimentan antes de aprobarlo. El proceso termina con el Documento de Diseño aprobado, con el que el equipo inicia el segundo semestre.

3.2.1. PROPUESTA DE INNOVACIÓN Y DEFINICIÓN DEL PROBLEMA

Objetivo: Formalizar una propuesta de proyecto sustentada en un problema u oportunidad real, un usuario o cliente definido y una propuesta de valor, para someterla a evaluación en la Licitación.

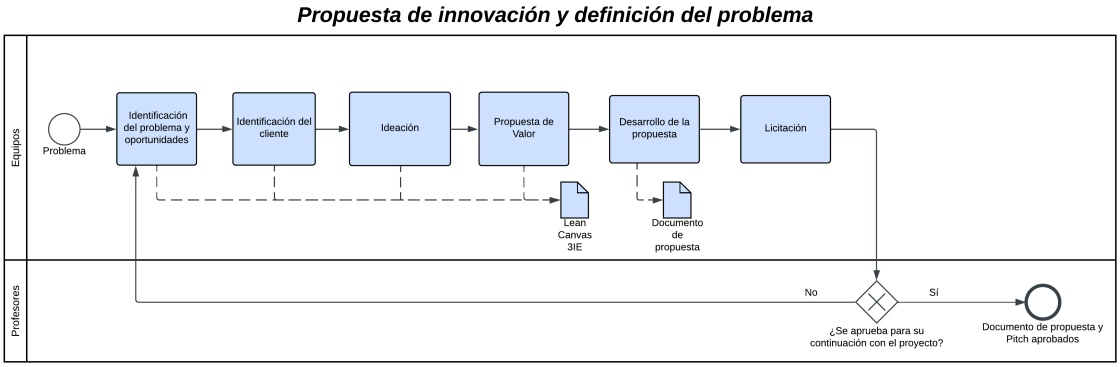


Figura 7: Proceso de propuesta de innovación y definición del problema. Fuente: Elaboración propia.

Entrada: Problemas u oportunidades identificados por el equipo.

Salida: Documento de propuesta y pitch aprobados.

Reglas de negocio:

1. Las cuatro etapas de innovación se recorren en el orden definido por Cayo [2], y cada una alimenta el Lean Canvas del equipo.
2. El Lean Canvas se construye con las plantillas del 3IE.
3. El documento de propuesta se desarrolla a partir del Lean Canvas y se entrega para una sola idea de proyecto.
4. Si la propuesta no se aprueba en la Licitación y la idea no tiene forma de ajustarse, el equipo vuelve a la identificación del problema y oportunidades y repite el proceso.

Descripción:

Este proceso agrupa las etapas con que el equipo construye y defiende su idea de proyecto (ver Figura 7)[10]. Su punto de partida no es una idea ya decidida, sino un conjunto de problemas u oportunidades que el equipo trae al comenzar el semestre. Esto no es nuevo en la Feria de Software. Calvo [1] ya reconocía que un equipo podía llegar con más de una idea de proyecto, y proponía que durante el llamado a postulación cada equipo presentara semanalmente ante los profesores cada una de sus ideas, para recibir su aprobación, observaciones o rechazo antes de formalizar una de ellas en el documento de postulación. Esta propuesta conserva esa lógica de partir desde varias alternativas.

Esa forma de avanzar es la que estructura el marco de innovación de Cayo [2], en el que se apoyan las cuatro primeras etapas de este proceso (ver Figura 3). Cayo organiza cada una de sus fases siguiendo la lógica del modelo del Doble Diamante, desarrollado por el British Design Council en 2005, que representa el proceso de diseño como una alternancia entre fases divergentes, en las que se abren opciones, y fases convergentes, en las que se sintetiza lo encontrado y se elige [2]. En su marco, cada fase se compone de una etapa de exploración (divergencia) y una de definición (convergencia), y termina en un resultado concreto que sirve de entrada a la siguiente [2]:

- **Identificación del problema y oportunidades** (explorar problemas y definir problema): el equipo explora la mayor cantidad de problemas que le permita el tiempo. Luego los describe, distingue sus causas de sus efectos y selecciona el más relevante con base en datos. Cayo [2] indica explícitamente que los equipos con más de una idea deben identificar los problemas asociados a cada una y elegir el de mayor impacto.
- **Identificación del cliente** (empatizar y definir usuario y mercado): el equipo busca comprender a su cliente ideal o early adopter: qué hace, qué lo frustra y qué resultados espera.
- **Ideación** (idear y priorizar): el equipo genera la mayor cantidad posible de ideas de solución y luego las selecciona y prioriza según su novedad, su impacto, su factibilidad dentro del curso y su capacidad de responder a los dolores del usuario.
- **Propuesta de valor** (diseñar y validar valor): el equipo analiza a su competencia y formula una propuesta de valor que explica cómo su solución resuelve el problema, en qué se diferencia de esas alternativas y qué valor entrega al cliente definido.

El recorrido entre estas fases no es estrictamente lineal. Como muestran las flechas punteadas de la Figura 3, lo que el equipo descubre en una fase puede llevarlo a revisar lo definido en una anterior, y desde la Licitación es posible volver a cualquiera de ellas. Cayo [2] diseñó este marco con una duración aproximada de seis semanas, de las cuales las dos primeras ocurren al final de la asignatura Sistemas de Gestión, donde se concentra la definición del problema, y las cuatro restantes durante GPI, donde se concentra la definición

[10] En los diagramas de procesos de este capítulo, las actividades en azul corresponden a tareas asociadas a innovación y emprendimiento, y las actividades en amarillo a tareas de desarrollo o de avance académico del curso.

de la solución. Así, cuando el equipo comienza GPI, ya cuenta con problemas explorados y una idea más madura.

La propuesta no modifica estas fases. Lo que sí formaliza es, por un lado, el formato con que el equipo registra lo trabajado en ellas, mediante las plantillas de Lean Canvas del 3IE.

LEAN CANVAS 3IE

Objetivo: Registrar en un lienzo el problema, el cliente, la solución y la propuesta de valor del proyecto, para reunir en un solo lugar lo construido en las cuatro etapas de innovación.

Entrada: Problemas u oportunidades identificados por el equipo.

Salida: Lean Canvas.

Descripción:

Cada etapa de innovación se trabaja con una o más plantillas del Instituto Internacional para la Innovación Empresarial (3IE), el instituto de la UTFSM que apoya a emprendimientos de base tecnológica [59]. Cada plantilla guía al equipo con preguntas concretas y cierra con una frase de síntesis, que corresponde al momento de convergencia de su fase. Lo trabajado en la etapa queda registrado de forma breve y comparable entre equipos (ver Tabla 3). En conjunto, estas plantillas completan el Lean Canvas del proyecto [39], estos pueden examinarse con más detalle en la sección *Anexos*.

Etapa de innovación	**Plantilla 3IE**	**Qué deja definido**
Identificación del problema y oportunidades	Template del Problema	Quién vive el problema, qué brecha enfrenta y qué consecuencias genera, sin incorporar todavía la solución.
Identificación del cliente	Template de Segmento de Clientes y Template de Cliente Ideal	El segmento prioritario, su tamaño (mercado total, atendible y alcanzable) y el perfil del cliente ideal.
Ideación	Template de la Solución	Qué entregará el equipo, su beneficio principal y el alcance de una primera versión.
Propuesta de valor	Template de Propuesta de Valor	Cómo la solución responde a los trabajos, dolores y ganancias del cliente, y en qué se diferencia de las alternativas actuales.

Tabla 3: Plantillas del 3IE utilizadas en cada etapa de innovación. Fuente: Elaboración propia a partir de las plantillas del 3IE.

DOCUMENTO DE PROPUESTA Y LICITACIÓN

Objetivo: Formalizar en un documento la idea de proyecto más sólida del equipo, para defenderla ante los evaluadores de la asignatura.

Entrada: Lean Canvas 3IE.

Salida: Documento de propuesta y pitch aprobados.

Descripción:

Aunque el proceso parte desde varios problemas, el documento de propuesta se entrega para una sola idea. En el marco vigente, este documento es el que define inicialmente la idea que pretende desarrollar cada equipo, y reúne el problema identificado y la solución propuesta, a quiénes está dirigido el producto, por qué es importante para ellos y en qué se diferencia de lo que ya existe en el mercado [1]. En esta propuesta, el documento se complementa directamente con el Lean Canvas 3IE, que ya reúne estos mismos elementos.

Con el documento listo, el equipo presenta su idea en la Licitación. La Licitación corresponde a la instancia en que cada equipo defiende su idea de proyecto, tal como la define el marco vigente al cierre de la Postulación de Proyectos [1]. A partir del documento de propuesta, el equipo presenta su proyecto ante los profesores.

El cierre de la Licitación es el punto de decisión más importante del proceso, porque define con qué proyecto trabajará el equipo durante el resto del año. Entre las críticas que Calvo [1] recoge de los legados de los estudiantes aparece, justamente, que trabajos aceptados al inicio del año fueron rechazados en el segundo semestre. El punto de decisión puede tener tres resultados:

- **La propuesta se aprueba**: el proyecto queda autorizado para continuar y el equipo pasa al proceso de definición del alcance y riesgos.
- **La propuesta no se aprueba, pero la idea puede ajustarse**: el equipo corrige su propuesta y la vuelve a presentar. Este caso no se representa en la Figura 7 porque no ocurre siempre, sino como un caso especial que los profesores resuelven según cada equipo. Corresponde al periodo breve de reformulación que el marco vigente ya contempla [1].
- **La propuesta no se aprueba y la idea no tiene forma de ajustarse**: el equipo vuelve a la identificación del problema y oportunidades y repite el proceso rápidamente con otra de las ideas que exploró, hasta llegar a una propuesta que pueda salir adelante.

3.2.2. DEFINICIÓN DEL ALCANCE Y RIESGOS

Objetivo: Acotar el alcance del proyecto según su naturaleza y reducir el riesgo de que el equipo no sea capaz de construir lo que propuso, para planificar el segundo semestre sobre una base viable.

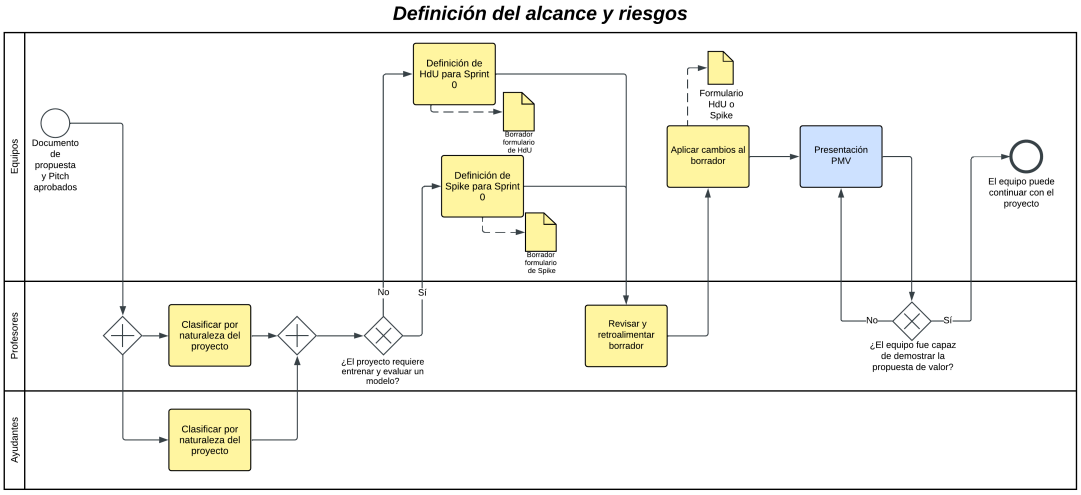


Figura 8: Proceso de definición del alcance y riesgos. Fuente: Elaboración propia.

Entrada: Documento de propuesta y pitch aprobados.

Salida: Clasificación de la naturaleza del proyecto, Formulario de Historias de Usuario o Formulario de Spike según corresponda y PMV presentado que demuestra la propuesta de valor del proyecto.

Reglas de negocio:

1. La clasificación es binaria y depende únicamente de si el proyecto requiere entrenar y evaluar un modelo de Machine Learning.
2. La clasificación no separa proyectos de software de proyectos de Machine Learning. Los proyectos con modelo propio mantienen el desarrollo de historias de usuario y agregan las tareas de CRISP-DM.
3. La actividad se llama Sprint 0 para ambas naturalezas de proyecto para conservar el lenguaje que se usa dentro del curso.
4. El Spike del Sprint 0 no se cuenta dentro de los dos Spikes que la regla vigente permite durante los Sprints 1 y 2.
5. Los profesores revisan y retroalimentan el borrador del formulario antes de que el equipo aplique los cambios y presente su PMV.
6. El PMV debe demostrar la propuesta de valor comprometida en la Licitación. Si el equipo no logra demostrarla, debe volver a presentarlo antes de continuar con el proyecto.

Descripción:

Una vez aprobada la propuesta en la Licitación, el equipo ya sabe qué problema quiere resolver y para quién, pero todavía no ha demostrado que es capaz de construir la solución que propuso. Este proceso existe para cerrar esa brecha antes de que el equipo planifique el segundo semestre (ver Figura 8). Para ello, el proceso responde dos preguntas en orden: qué tipo de proyecto es, y si el equipo puede entregar lo que comprometió.

La primera pregunta se resuelve con la clasificación de la naturaleza del proyecto, que profesores y ayudantes realizan en paralelo y luego resuelven en conjunto. Esa clasificación define cómo el equipo abordará el Sprint 0 en el caso de ser un proyecto sin modelo propio, será mediante historias de usuario, o si entrena y evalúa un modelo propio, mediante un Spike. En ambos casos, el equipo prepara un borrador de su formulario, los profesores lo revisan y retroalimentan, y el equipo aplica los cambios antes de presentar.

La segunda pregunta se resuelve en la presentación del PMV, que es donde hay un cambio respecto del marco vigente que se lleva aplicando informalmente. Calvo [1] planteaba el prototipo inicial en torno a historias de alto valor, pero de menor dificultad. Esta propuesta, en cambio, formaliza la idea de que el Sprint 0 aborde las historias críticas del proyecto, aquellas que materializan su propuesta de valor y concentran sus mayores riesgos, de modo que la presentación funcione como un control de riesgo, si el equipo no logra demostrar su propuesta de valor, vuelve a presentar antes de continuar haciendo los cambios correspondientes. Las secciones siguientes detallan cada una de estas actividades.

CLASIFICACIÓN POR NATURALEZA DEL PROYECTO

Este proceso parte con una actividad nueva, situada inmediatamente después de la Licitación, cuyo objetivo es evaluar la naturaleza del proyecto y determinar si seguirá el camino de los proyectos sin entrenamiento de modelo propio o el de los proyectos con entrenamiento y evaluación de modelo propio. En adelante, ambos caminos se denominan, de forma abreviada, camino sin modelo propio y camino con modelo propio. La clasificación depende de un único criterio, si el desarrollo del proyecto requiere entrenar y evaluar un modelo de Machine Learning. Si lo requiere, el proyecto sigue el camino con modelo propio; si no, sigue el camino sin modelo propio. Como se fundamentó en el marco conceptual, este criterio es más acotado que otros posibles, ya que no basta con que el proyecto use datos ni con que se enmarque, de forma general, dentro de la Ciencia de Datos o la Inteligencia Artificial, porque lo que justifica un camino distinto es el ciclo de vida propio que exige un modelo, no el área temática del proyecto.

Ambos caminos no son excluyentes en cuanto al desarrollo de software. Todo proyecto de la Feria construye un producto que su usuario utiliza y, por lo tanto, todos desarrollan historias de usuario durante el segundo semestre. Lo que distingue a los proyectos con modelo propio es que, además de ese desarrollo, incorporan las actividades del ciclo de vida del modelo según CRISP-DM. Un proyecto que construye una plataforma web completa y, al mismo tiempo, entrena un modelo propio sigue, por lo tanto, el camino con modelo propio, en el que el desarrollo del software continúa en paralelo a las tareas de modelado. El segundo camino no reemplaza al primero, sino que lo amplía.

La clasificación no queda a criterio de un solo evaluador ni se decide de forma automática. Profesores y ayudantes clasifican el proyecto en paralelo, en base al documento de propuesta y al pitch ya aprobados en la Licitación, y luego resuelven en conjunto la

clasificación final. De esta manera, la clasificación no depende únicamente del equipo de estudiantes ni de la apreciación individual de un ayudante, sino de un acuerdo entre quienes evaluarán el proyecto durante el resto del proceso.

A partir de esa clasificación, el equipo define el contenido con el que abordará el Sprint 0. En el caso de un proyecto sin modelo propio, lo abordará mediante historias de usuario. Para el caso de un proyecto con modelo propio, en cambio, lo abordará mediante un Spike que debe cumplir con los lineamientos definidos en la sección *Sprint 0: proyectos con modelo propio*. El nombre de la actividad (Sprint 0) se mantiene igual para ambos caminos, evitando así confusiones entre los estudiantes y en los porcentajes de evaluación del curso.

SPRINT 0: PROYECTOS SIN MODELO PROPIO

Objetivo: Definir las historias de usuario críticas del proyecto, para mitigar sus principales riesgos y demostrar su propuesta de valor en la presentación del PMV.

Entrada: Proyecto clasificado en el camino sin modelo propio.

Salida: Formulario de Historias de Usuario, con la retroalimentación de los profesores aplicada.

Descripción:

Para los proyectos sin modelo propio, el Sprint 0 corresponde a lo definido por Calvo [1] dentro de la categoría de Realización del Prototipo (ver Figura 2), y conserva su propósito formativo, dar a los alumnos un primer acercamiento al producto y experiencia con las habilidades de gestión de proyectos y herramientas técnicas que usarán durante el resto del proyecto. El equipo define las historias de usuario del Sprint 0 en un borrador del formulario, que los profesores revisan y retroalimentan. Con esa retroalimentación, el equipo aplica los cambios al formulario y llega a la presentación del PMV con esas historias.

La diferencia respecto del marco vigente está en el criterio con que se eligen estas historias. Calvo [1] las definía como historias de alto valor para el cliente o usuario, pero de menor dificultad frente a otras igual de importantes. Esta propuesta, en cambio, formaliza la petición de que el Sprint 0 aborde las historias críticas del proyecto, las que materializan su propuesta de valor y, por lo mismo, concentran sus mayores riesgos, ya sea porque dependen de una tecnología que el equipo no domina, de una integración externa o de una funcionalidad que nunca ha construido. Dejar estas historias para el segundo semestre, como ocurre cuando se priorizan las de menor dificultad, significa descubrir tarde si el equipo es capaz de construir lo que propuso. Abordarlas en el Sprint 0, en cambio, permite mitigar esos riesgos al inicio del proyecto y registrar los que persistan en la lista de riesgos técnicos del Documento de Diseño.

SPRINT 0: PROYECTOS CON MODELO PROPIO

Objetivo: Determinar si los datos disponibles permiten resolver el problema planteado, para reducir la incertidumbre técnica del proyecto antes de comprometer el desarrollo del segundo semestre.

Entrada: Proyecto clasificado en el camino con modelo propio.

Salida: Formulario de Spike, con la retroalimentación de los profesores aplicada.

Descripción:

Para los proyectos con modelo propio, el Sprint 0 se ejecuta de manera distinta. En lugar de historias de usuario, el equipo desarrolla un Formulario de Spike (equivalente adaptado del Formulario de Historias de Usuario que hoy se entrega en esta misma instancia para el camino sin modelo propio), extendiendo el alcance habitual de un Spike (definido en el marco conceptual como un ítem acotado del backlog) a una etapa completa del proyecto. Su objetivo no es demostrar una funcionalidad terminada, sino reducir la incertidumbre técnica del proyecto antes de comprometer el desarrollo del resto del semestre. Como se estableció en el marco conceptual, un proyecto de ML no puede asumir que cada ciclo de trabajo produce un incremento de producto evaluable visiblemente, gran parte de su esfuerzo inicial se concentra en determinar si los datos disponibles permiten resolver el problema planteado.

Para cumplir ese objetivo, el Spike no recorre CRISP-DM completo, sino solo las fases que permiten responder si el proyecto es viable: la comprensión del negocio, la comprensión de los datos y la primera tarea de la preparación de los datos, que consiste en seleccionar qué datos se usarán [3]. Todo esto ocurre dentro del mismo plazo que hoy tiene el desarrollo del prototipo inicial. La construcción, integración y formateo del resto de los datos, la evaluación formal de historias de usuario y el desarrollo de un prototipo interactivo funcional quedan para los sprints del segundo semestre, una vez que el proyecto cuenta con evidencia de que es viable.

Lo que el equipo trabaja en el Spike es el insumo de lo que presentará en el PMV. En un proyecto sin modelo propio, el riesgo principal es no ser capaz de construir las historias críticas. En un proyecto con modelo propio, se añade el riesgo de que los datos disponibles no permitan entrenar un modelo que cumpla lo prometido. Por eso, el equipo no llega al PMV a mostrar una funcionalidad, sino la evidencia con que responde a esos riesgos. Para que esto quede claro desde el inicio, el Spike se organiza en dos momentos:

- **Antes de comenzar, en el Formulario de Spike**: el equipo deja por escrito qué busca resolver con el modelo, cómo medirá si lo logró, con qué recursos cuenta y, sobre todo, qué riesgos debe despejar, formulados como preguntas concretas (por ejemplo, ¿existen suficientes datos?, ¿se puede acceder a ellos dentro del plazo?). Este borrador es el que los profesores revisan y retroalimentan.
- **Al finalizar, en la presentación del PMV**: el equipo presenta lo que encontró al explorar los datos, la calidad de esos datos, qué datos usará y cuáles descarta, y la respuesta a cada

una de las preguntas planteadas en el formulario, cerrando con una conclusión explícita de viabilidad.

La Tabla 4 propone la plantilla que ordena ambos momentos. Sus campos siguen las tareas que CRISP-DM define para cada fase [3], acotadas a lo que un equipo puede abordar dentro del Sprint 0.

Sección	Campos	Fase de CRISP-DM	Se entrega en
1. Objetivo de negocio	Problema que se busca resolver con el modelo, quién usará su resultado y qué decisión o tarea apoya.	Comprensión del negocio	Formulario
2. Objetivo del modelo	Qué debe hacer el modelo (clasificar, predecir, recomendar, etc.), con qué datos de entrada y qué resultado entrega.	Comprensión del negocio	Formulario
3. Criterio de éxito	Criterio con que se evaluará el modelo y el umbral mínimo que debe alcanzar.	Comprensión del negocio	Formulario
4. Inventario de recursos	Fuentes de datos disponibles (origen, forma de acceso, volumen estimado y si contienen datos personales), herramientas y tecnologías, y cantidad.	Comprensión del negocio	Formulario
5. Riesgos y preguntas a resolver	Principales riesgos del proyecto, formulados como preguntas que el Spike debe responder, junto con cómo se verificará cada una.	Comprensión del negocio	Formulario
6. Plan del Spike	Actividades y plazos dentro del equipo para el Sprint 0.	Comprensión del negocio	Formulario
7. Descripción y exploración de los datos	Volumen, variables disponibles, distribución y principales hallazgos.	Comprensión de los datos	PMV
8. Calidad de los datos	Datos faltantes, errores, posibles sesgos y disponibilidad de etiquetas.	Comprensión de los datos	PMV
9. Selección de los datos	Qué datos se usarán y cuáles se descartan, con su justificación.	Preparación de los datos	PMV
10. Respuesta a los riesgos y viabilidad	Respuesta, con evidencia, a cada pregunta de la sección 5, y conclusión explícita de viabilidad: si los datos permiten construir el modelo comprometido y, si no, qué ajuste de alcance propone el equipo.	Síntesis del Spike	PMV

Tabla 4: Plantilla propuesta para el Formulario de Spike y la presentación del PMV en proyectos con modelo propio. Fuente: Elaboración propia a partir de las tareas de CRISP-DM.

Al igual que el borrador de historias de usuario en el camino sin modelo propio, el borrador del Formulario de Spike es revisado y retroalimentado por los profesores antes de que el

equipo aplique los cambios. El criterio de éxito definido en la sección 3 es, además, el que el Documento de Diseño registra formalmente y el que se evalúa en el punto de decisión al cierre del Sprint 2. Así, el Spike no deja solo una intención declarada en la Licitación, sino una base realista que permite planificar el resto del proyecto en el Documento de Diseño.

PRESENTACIÓN DEL PMV

Objetivo: Demostrar la propuesta de valor comprometida en la Licitación, para comprobar que el equipo es capaz de entregarla antes de continuar con el proyecto.

Entrada: Formulario de Historias de Usuario o Formulario de Spike, con la retroalimentación de los profesores aplicada.

Salida: PMV presentado. Si demuestra la propuesta de valor, el equipo puede continuar con el proyecto.

Descripción:

En el marco vigente, la presentación del prototipo se centra en las historias de usuario aptas para un prototipo, es decir, las de alto valor pero de menor dificultad [1]. Con ese criterio, un equipo puede presentar un prototipo que funciona sin haber demostrado que es capaz de construir aquello que hace distinto a su proyecto. En la práctica, el curso ya ha ido cambiando esto y ha pedido que esta instancia sea la encargada de la mitigación de riesgos, y esta propuesta lo formaliza al exigir que el PMV demuestre la propuesta de valor del proyecto. Este criterio coincide además con el programa de GPI, que aún no entra en vigencia, elaborado para el nuevo plan de estudios en el que se espera que el equipo sea capaz de «mitigar el mayor riesgo de su proyecto mediante el desarrollo de un producto mínimo viable (PMV) que garantiza la viabilidad técnica del desarrollo de la idea» [14]. Así, la presentación funciona como un control de riesgo, ya que antes de planificar el segundo semestre en el Documento de Diseño, los profesores verifican que el equipo es capaz de hacer lo que dice.

Lo que se presenta depende de la naturaleza del proyecto. En los proyectos sin modelo propio, el equipo demuestra las historias críticas desarrolladas en el Sprint 0, que materializan su propuesta de valor, en una instancia de Sprint Review complementada con un Sprint Retrospective interno del equipo. En los proyectos con modelo propio, el equipo demuestra su propuesta de valor mediante la evidencia de viabilidad recogida en el Spike, es decir, las secciones 7 a 10 de la plantilla del Formulario de Spike (ver Tabla 4), que consisten en la exploración y calidad de los datos, la selección de qué datos se usarán y cuáles se descartan, y la respuesta a cada riesgo planteado, cerrando con una conclusión explícita de viabilidad. En este caso, la Retrospective no se centra en historias de usuario completadas, sino en el proceso de investigación realizado durante el Spike.

3.2.3. PLANIFICACIÓN Y DISEÑO

Objetivo: Consolidar el modelo de negocios del proyecto y definir en el Documento de Diseño qué se va a construir y cómo, para que ambos sirvan de base al desarrollo del segundo semestre.

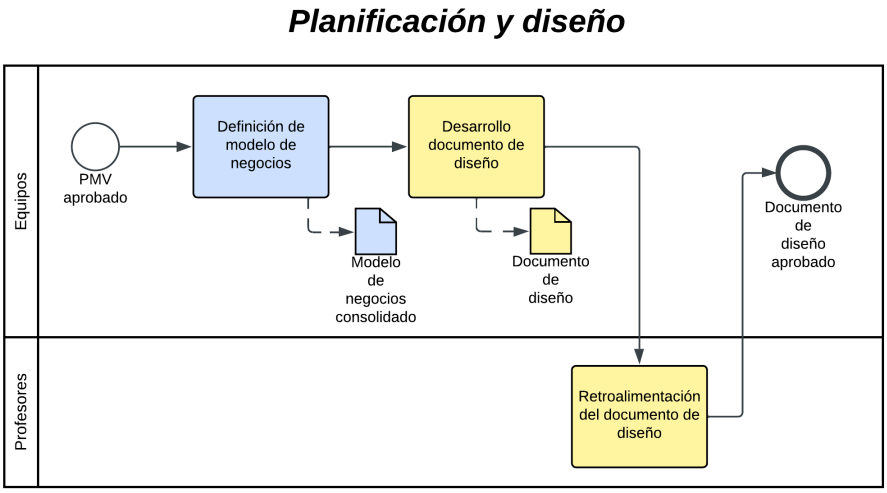


Figura 9: Proceso de planificación y diseño. Fuente: Elaboración propia.

Entrada: PMV aprobado.

Salida: Modelo de negocios consolidado y Documento de Diseño aprobado.

Reglas de negocio:

1. El modelo de negocios consolidado incluye el Business Model Canvas completo y una meta de clientes contrastada con el mercado alcanzable.
2. El Documento de Diseño incluye las secciones de diagrama de arquitectura, modelo de base de datos, atributos de calidad y uso de IA en el desarrollo.
3. En los proyectos con modelo propio, el Documento de Diseño registra el criterio de éxito del modelo.
4. Los profesores retroalimentan el Documento de Diseño antes del término del semestre.

Descripción:

Una vez aprobado el PMV, el equipo ya demostró que es capaz de construir lo que propuso. Lo que falta antes de iniciar el segundo semestre es convertir esa evidencia en un plan, tanto del negocio como del producto. Este proceso cierra el primer semestre con esa planificación y se compone de dos entregables (ver Figura 9).

El primero es el modelo de negocios consolidado, en el que el equipo completa su Business Model Canvas. El segundo es el Documento de Diseño, que reemplaza al actual Plan de Proyecto y es el entregable que más cambia respecto del marco vigente. Además de planificar qué se va a construir, fija cómo se va a construir y obliga al equipo a justificar

sus decisiones de diseño, lo que responde directamente a la condición del uso de IA como herramienta de desarrollo descrita en el marco conceptual. Los profesores retroalimentan este documento antes del término del semestre, de modo que el equipo inicia el Sprint 1 con un plan revisado.

MODELO DE NEGOCIOS CONSOLIDADO

Objetivo: Consolidar en un solo entregable cómo funciona el negocio del proyecto y cuántos clientes necesita para ser viable, para iniciar el segundo semestre con un negocio claro.

Entrada: Lean Canvas 3IE, con las plantillas trabajadas en el proceso de propuesta de innovación.

Salida: Modelo de negocios consolidado, con sus cuatro secciones (oportunidad de negocio, producto o servicio, mercado y modelo de negocio), incluida la meta de clientes.

Descripción:

Durante el primer semestre, el trabajo de emprendimiento funciona como preparación para el proceso de emprendimiento del segundo semestre. Después de la presentación del PMV, el equipo consolida su modelo de negocios, que tiene como centro el Business Model Canvas [35]. Mientras el Lean Canvas trabajado al inicio del semestre se concentra en el problema, el cliente y la propuesta de valor, el Business Model Canvas agrega cómo opera el negocio (actividades, recursos y socios clave) y cómo se relaciona con sus clientes.

Hoy, el Business Model Canvas se solicita recién después del segundo sprint del segundo semestre, junto con un modelo de negocio preliminar que se limita a justificar los bloques del lienzo y a completar una planilla de ingresos y costos [38]. Adelantarlo al cierre del primer semestre responde a dos razones. La primera es que se trata de definiciones que el equipo necesita tener resueltas antes de construir su marca y su Plan de Marketing, por lo que no tiene sentido pedirlas cuando ambos ya deberían estar en marcha. La segunda es que, en este punto, el equipo ya cuenta con la experiencia del PMV, lo que le permite definir con más realismo qué actividades y recursos requiere su solución.

El apellido «consolidado» indica que el entregable va más allá del lienzo. Reúne en un mismo documento lo que el equipo ya trabajó en las plantillas del Lean Canvas, el Business Model Canvas completo y una meta de clientes, organizados en las cuatro secciones que propone la Tabla 5. La meta de clientes definida aquí es, además, el punto de partida del Plan de Marketing del segundo semestre, que establece cómo alcanzarla.

Sección	Contenido mínimo	Se construye a partir de
Oportunidad de negocio	Fenómeno que origina la oportunidad y factores de viabilidad y urgencia, junto con la evidencia que la respalda.	Template del Problema
Producto o servicio	Características, funcionalidades y atributos, tabla de benchmarking con soluciones cercanas y diferenciación.	Template de la Solución y Template de Propuesta de Valor
Mercado	Estimación de mercado total, atendible y alcanzable, con su fuente o supuesto.	Template de Segmento de Clientes
Modelo de negocio	Business Model Canvas completo y meta de clientes contrastada con el mercado alcanzable.	Lean Canvas 3IE y resultados del PMV

Tabla 5: Plantilla propuesta para el modelo de negocios consolidado. Fuente: Elaboración propia.

Las tres primeras secciones no parten de cero, ya que retoman lo que el equipo trabajó en las plantillas del 3IE durante el proceso de propuesta de innovación y lo trabajado para el documento de propuesta durante el proceso de definición de alcance y riesgos. Lo nuevo se concentra en la cuarta sección.

DOCUMENTO DE DISEÑO

Objetivo: Definir qué se va a construir y cómo, incluyendo las decisiones de arquitectura y diseño técnico del sistema, para que sirva de base a los sprints del segundo semestre.

Entrada: Formulario de Historias de Usuario o Formulario de Spike, junto con la clasificación de naturaleza del proyecto.

Salida: Documento de Diseño, entregado formalmente al cierre del primer semestre.

Descripción:

Una vez aprobado el PMV y consolidado el modelo de negocios, el primer semestre culmina con una instancia de planificación tentativa que hoy se materializa en la entrega denominada Plan de Proyecto. Esta propuesta plantea renombrar dicha entrega a Documento de Diseño, extendiendo su contenido más allá de lo que hoy cubre.

El nuevo nombre no es solo una etiqueta, responde a algo que el documento actual no cubre. El Plan de Proyecto actual está orientado casi en su totalidad a la planificación del producto desde una perspectiva de negocio y de gestión (resumen del proyecto, segmento de usuarios, propuesta de valor, riesgos técnicos, distribución tentativa de historias de

usuario), pero no contempla un espacio explícito para las decisiones de arquitectura y diseño técnico del sistema que se va a construir. Esa ausencia es, precisamente, el mismo punto de control identificado en el marco conceptual sobre la «trampa de la implementación». Sin un documento donde estas decisiones deban plasmarse desde el inicio, quedan implícitas en el código, en particular en un contexto donde buena parte del desarrollo se apoya en herramientas de generación de código asistida por IA [43]. Llamar al documento «Documento de Diseño» en lugar de «Plan de Proyecto» busca dejar explícito, desde el nombre mismo de la entrega, que a partir de esta versión el documento no solo planifica qué se va a construir, sino que también fija el cómo, sentando además la base sobre la que se apoya la actividad de actualización de documentación que se repite en los tres sprints del segundo semestre.

En términos de contenido, el Documento de Diseño conserva las secciones ya existentes en el Plan de Proyecto, que son el resumen del proyecto, la descripción de los tipos de usuario y la Lista de riesgos técnicos con su plan de mitigación. A estas secciones se agregan cuatro nuevas, comunes a ambas naturalezas de proyecto. En cada una, el equipo no solo describe lo que decidió, sino que justifica por qué lo decidió. Esa justificación es la que permite revisar más adelante si una decisión sigue siendo válida cuando el desarrollo avanza, y es también la que evita que las decisiones queden implícitas en el código, en particular cuando parte de ese código se genera con IA [43].

- **Diagrama de arquitectura**: representación de los principales componentes del sistema y de cómo se relacionan entre sí, junto con la justificación de por qué se eligió esa arquitectura frente a otras alternativas, en relación con los atributos de calidad.
- **Modelo de base de datos**: representación de las entidades que el sistema almacena, con sus atributos y relaciones. El equipo la define antes de avanzar en la implementación y la mantiene actualizada a medida que cambia el alcance del proyecto.
- **Atributos de calidad**: los atributos de calidad priorizados para el proyecto (por ejemplo, usabilidad, rendimiento o seguridad), que hasta ahora no contaban con un espacio explícito de registro. Para cada uno, el equipo justifica por qué es prioritario en relación con sus usuarios y su propuesta de valor.
- **Uso de IA en el desarrollo**: declaración de las herramientas de IA que el equipo planea usar, con qué nivel de autonomía (autocompletado asistido, vibe coding o agentic coding, según la distinción del marco conceptual) y para qué tipo de tareas. No busca restringir su uso, sino dejar explícito desde el inicio cuánto del código será generado por un modelo y, por lo tanto, cuánta revisión requerirá. El equipo justifica por qué usará cada nivel de autonomía en cada tipo de tarea y cómo revisará lo que se genere. Además, declara qué datos del proyecto compartirá con esas herramientas, en particular si alguno corresponde a datos personales o sensibles, ya que el uso de servicios externos de IA durante el desarrollo puede llevar a que esa información salga del sistema sin que el equipo lo evalúe.

Adicionalmente, la sección que hoy existe como «Herramientas y tecnología a utilizar y Requisitos no funcionales» se separa en dos secciones independientes, una de Herramientas

y tecnología y otra dedicada exclusivamente a los Requisitos no funcionales del sistema, sin cambios en su contenido. La Tabla 6 resume la estructura propuesta del documento.

Sección	**Qué contiene**	**Qué debe justificar**
Resumen del proyecto	Descripción del proyecto y su propuesta de valor.	Sin cambios respecto del Plan de Proyecto.
Tipos de usuario	Usuarios que interactúan con el sistema.	Sin cambios respecto del Plan de Proyecto.
Lista de riesgos técnicos	Riesgos técnicos con su plan de mitigación.	Sin cambios respecto del Plan de Proyecto.
Atributos de calidad	Atributos priorizados y cómo se verificará cada uno.	Por qué esos atributos son prioritarios para los usuarios y la propuesta de valor.
Diagrama de arquitectura	Componentes del sistema y sus relaciones.	Por qué esa arquitectura y no otra, en relación con los atributos de calidad.
Modelo de base de datos	Entidades, atributos y relaciones.	Tipo de almacenamiento y decisiones de estructura que no sean evidentes.
Uso de IA en el desarrollo	Herramientas, nivel de autonomía, tipo de tareas y datos del proyecto que se comparten con ellas.	Por qué ese nivel de autonomía en cada tarea, cómo se revisará lo generado y por qué es seguro compartir esos datos.
Herramientas y tecnología	Lenguajes, frameworks, servicios y plataformas.	Sin cambios respecto del Plan de Proyecto.
Requisitos no funcionales	Requisitos no funcionales del sistema.	Sin cambios respecto del Plan de Proyecto.
Distribución tentativa	Historias de usuario por sprint o, en proyectos con modelo propio, distribución combinada (ver Tabla 7).	Sin cambios respecto del Plan de Proyecto en proyectos sin modelo propio. En proyectos con modelo propio, se justifica por qué ese orden según valor y riesgo.
Criterio de éxito del modelo	Solo en proyectos con modelo propio. Métrica y umbral definidos en el Formulario de Spike.	Se justifica en el Formulario de Spike.

Tabla 6: Plantilla propuesta para el Documento de Diseño. Fuente: Elaboración propia a partir del Plan de Proyecto del marco vigente.

La sección de distribución tentativa de historias de usuario, en cambio, sí se bifurca según la naturaleza del proyecto, de manera consistente con los Sprints 1, 2 y 3 que se describen en la sección siguiente. Para los proyectos sin modelo propio, esta sección se mantiene sin

cambios y corresponde a una distribución de las historias de usuario a lo largo de los tres sprints del segundo semestre, ordenadas por valor. Para los proyectos con modelo propio, en cambio, esta sección pasa a ser una distribución combinada de historias de usuario y tareas de CRISP-DM. Como esta forma de planificar es nueva para el curso, la Tabla 7 propone qué se espera en cada sprint, a partir de las tareas que CRISP-DM define para sus fases [3].

Sprint	**Fases de CRISP-DM**	**Tareas de modelado esperadas**	**Historias de usuario**	**Evidencia al cierre**
Sprint 1	Preparación de los datos y modelado (fases 3 y 4)	Limpiar, construir, integrar y formatear los datos seleccionados en el Spike. Elegir la técnica de modelado, diseñar cómo se probará y construir una primera versión del modelo.	Historias que no dependen del resultado del modelo, como la interfaz o la gestión de usuarios.	Datos preparados y una primera versión del modelo con su desempeño medido.
Sprint 2	Evaluación y despliegue (fases 5 y 6)	Evaluar el modelo frente al criterio de éxito definido en el Formulario de Spike y planificar su despliegue y monitoreo dentro del producto.	Historias que integran el resultado del modelo en el producto.	Modelo evaluado frente al criterio de éxito, que alimenta el punto de decisión al cierre del sprint y funcionalidades principales en relación con el modelo completas.
Sprint 3	Sin fases pendientes	Sin tareas de modelado de fondo, aunque pueden quedar ajustes al modelo, en particular si el punto de decisión del Sprint 2 redujo su alcance.	Igual que en proyectos sin modelo propio.	Incremento del producto con el modelo integrado.

Tabla 7: Distribución propuesta del trabajo en proyectos con modelo propio. Fuente: Elaboración propia a partir de las tareas de CRISP-DM.

Esta propuesta no fija de antemano qué proporción de la capacidad de cada sprint se destina a tareas de modelado y cuál a historias de usuario, ya que la distribución se acuerda entre profesores y estudiantes según la cantidad de integrantes del equipo. Como punto de partida, se sugiere que en los Sprints 1 y 2 las tareas de modelado ocupen cerca del 40% de la capacidad del equipo y las historias de usuario el 60% restante, de modo que el sprint

siempre deje un incremento de producto evaluable aunque el modelado no avance como se esperaba. El equipo propone su distribución en el Documento de Diseño y la acuerda con los profesores durante la retroalimentación.

Por último, para los proyectos con modelo propio, el Documento de Diseño es donde se formaliza por escrito el criterio de éxito del modelo que determina, al cierre del Sprint 2, si el objetivo de Machine Learning se cumplió o si su alcance debe reducirse para el Sprint 3. Este criterio no se define recién en este documento. Como se estableció en el Sprint 0, ya forma parte del contenido del Formulario de Spike, dentro de los criterios de éxito definidos en la comprensión del negocio (fase 1 de CRISP-DM, ver *Sprint 0: proyectos con modelo propio*). Lo que aporta el Documento de Diseño es dejarlo registrado formalmente junto con el resto de la planificación del proyecto, disponible como referencia explícita para el punto de decisión que lo evalúa al cierre del Sprint 2.

Antes de su aprobación, los profesores revisan y retroalimentan el documento, de modo que el equipo inicia el Sprint 1 con un Documento de Diseño aprobado.

3.3. ESTRUCTURA SEGUNDO SEMESTRE

Objetivo: Desarrollar el producto y construir en paralelo el proyecto como emprendimiento, para presentarlo en la Feria de Software y cerrar el año con una base para continuarlo.

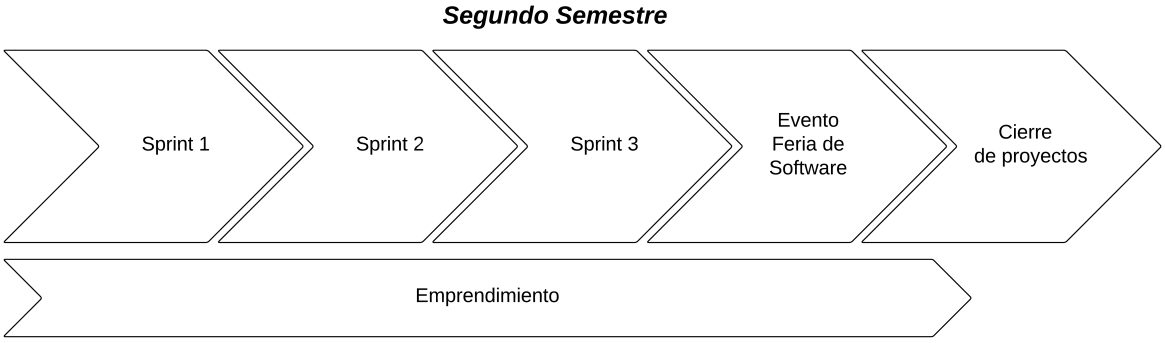


Figura 10: Procesos del segundo semestre. Fuente: Elaboración propia.

Descripción:

El segundo semestre se organiza en cinco procesos consecutivos y un proceso que corre en paralelo a ellos (ver Figura 10). Al igual que en el primer semestre, cada proceso deja un resultado concreto que sirve de entrada al siguiente, y no comienza hasta que el anterior se aprueba.

- **Sprints 1, 2 y 3**: el equipo desarrolla su producto en tres sprints que comparten una misma estructura para ambas naturalezas de proyecto. Las diferencias se concentran en el contenido de lo que se entrega en cada sprint y desaparecen en el Sprint 3.

- **Evento Feria de Software**: el equipo presenta su producto y su pitch como startup en el evento que da nombre al ciclo de asignaturas. La propuesta no modifica el evento, pero documenta el Demoday, que hoy lo diferencia de lo definido por Calvo [1].
- **Cierre de proyectos**: el equipo cierra el año con el Documento de Legado, un registro de las experiencias del año para las futuras generaciones, fundamentado en el PMBOK.
- **Emprendimiento**: proceso paralelo que comienza con el Sprint 1 y se extiende hasta después del evento de Feria de Software, documentado en la sección *Emprendimiento*.

Antes de detallar cada sprint conviene explicar qué tienen en común, porque esa estructura compartida es la que materializa la lógica de adecuación de esta propuesta. Los tres sprints siguen la misma secuencia. Comienzan con la entrega formal de lo que el equipo se compromete a desarrollar, continúan con un conjunto de actividades que ocurren en paralelo, y terminan con la Sprint Review, la Sprint Retrospective y la evaluación de profesores y ayudantes. Esta secuencia conserva las ceremonias, los roles y el calendario del marco vigente [1]. Lo que cambia de un sprint a otro es qué actividades se suman a esa base y, en los Sprints 1 y 2, el contenido de la entrega según la naturaleza del proyecto.

En los tres sprints, el cierre se compone de las mismas tres instancias. En la Sprint Review el equipo presenta el incremento desarrollado para recibir retroalimentación [13]. En la Sprint Retrospective, de carácter interno, el equipo revisa cómo fue su propio funcionamiento durante el sprint [13]. Finalmente, profesores y ayudantes evalúan el sprint en paralelo, ambos en su rol de Stakeholders definido en el marco vigente [1]. Esta evaluación ya existe hoy en el curso. Lo que hace la propuesta es dejarla explícita en el flujo como el paso que da por aprobado el sprint, de modo que el siguiente comience solo cuando el anterior está cerrado.

3.3.1. SPRINT 1: DESARROLLO INICIAL

Objetivo: Construir el primer incremento del producto a partir del Documento de Diseño y, en los proyectos con modelo propio, una primera versión del modelo, para dejar construida la base sobre la que se apoyan los sprints siguientes.

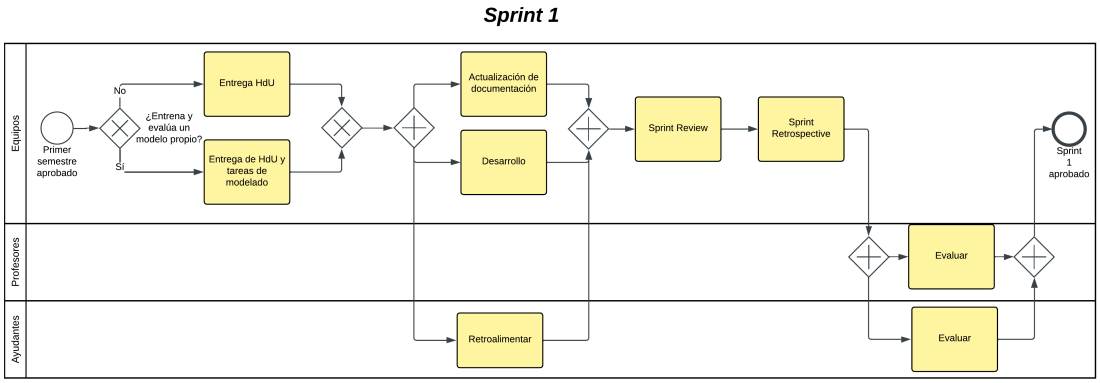


Figura 11: Proceso de Sprint 1. Fuente: Elaboración propia.

Entrada: Documento de Diseño aprobado al cierre del proceso de *Planificación y diseño*.

Salida: Incremento del producto presentado en la Sprint Review, Documento de Diseño actualizado y Sprint 1 aprobado por profesores y ayudantes.

Reglas de negocio:

1. La naturaleza del proyecto no se vuelve a evaluar. El gateway con que comienza el sprint solo enruta el flujo según la clasificación ya resuelta en el proceso de definición del alcance y riesgos.
2. Lo que el equipo compromete en la entrega viene de la distribución tentativa del Documento de Diseño. Si el equipo cambia algo respecto de esa distribución, lo justifica en la misma entrega.
3. La entrega del Sprint 1 debe abordar las historias de usuario esenciales e importantes del producto, es decir, las que sostienen su funcionamiento básico y de las que dependen las funcionalidades de los sprints siguientes.
4. En los proyectos con modelo propio, la entrega combina historias de usuario con tareas de las fases 3 y 4 de CRISP-DM. La proporción entre ambos tipos de trabajo se acuerda entre profesores y estudiantes según la cantidad de integrantes del equipo, y se sugiere como punto de partida cerca de un 40% de tareas de modelado y un 60% de historias de usuario.
5. Los Spikes que el equipo incluya en la entrega se rigen por la regla vigente del curso, con un máximo de dos durante todo el proyecto, solo en los Sprints 1 y 2 y con un tope de 30 puntos de historia cada uno [22].
6. El Desarrollo y la Actualización de documentación deben estar terminados antes de la Sprint Review.

Descripción:

El sprint se inicia con la entrega formal de lo planificado, esta entrega fija el conjunto de trabajo con el que el equipo se compromete para el ciclo, aunque en este caso ese compromiso deriva del Documento de Diseño ya revisado por los profesores. En los proyectos sin modelo propio, la entrega corresponde a las historias de usuario asignadas

al Sprint 1 en el Documento de Diseño, sin cambios respecto de cómo se ejecuta esta instancia hoy. Para ambas naturalezas, eso sí, el Sprint 1 debe abordar las historias de usuario esenciales e importantes del producto. Este Sprint es el momento de construir la base sobre la que se apoya el resto, es decir, las funcionalidades sin las cuales el producto no puede usarse y de las que dependen las historias de los sprints siguientes. Dejarlas para más adelante significaría construir funcionalidades sobre una base que todavía no existe, y llegar al Sprint 3 con lo esencial incompleto, que es justamente el sprint donde el equipo debería concentrarse en terminar el producto y en las funcionalidades deseables u opcionales. En los proyectos con modelo propio, en cambio, la entrega combina historias de usuario con tareas propias de las fases 3 (preparación de los datos) y 4 (modelado) de CRISP-DM, en un reparto de puntos de historia acordado con los profesores, para el que se sugiere cerca de un 40% de tareas de modelado y un 60% de historias de usuario (ver Tabla 7).

Este reparto no es una decisión arbitraria, sino la respuesta concreta a una tensión identificada en el marco conceptual. Scrum asume que cada sprint entrega un incremento evaluable, pero el trabajo de preparación y modelado de datos es exploratorio y no siempre produce algo demostrable al cierre del ciclo [13]. Si el sprint completo se dedicara solo a esta exploración, el equipo podría llegar a la Sprint Review sin nada que presentar en caso de que el modelado no avance a tiempo. Reservar la mayor parte de la capacidad del sprint a historias de usuario garantiza, en cambio, que el equipo siempre cuente con un incremento funcional que mostrar, independiente de cómo avance el trabajo de modelado. Esta forma de combinar CRISP-DM dentro de la cadencia de Scrum es consistente con lo que plantea la literatura sobre la integración de ambos marcos, que propone ajustar qué se entiende por «incremento» y por «hito» dentro de cada sprint, en lugar de abandonar Scrum para proyectos con modelo propio [48]. La proporción sugerida prioriza el trabajo con incremento visible y se ajusta según la cantidad de integrantes del equipo, porque de ella depende cuántas personas pueden dedicarse al modelado sin descuidar el producto.

A partir de la entrega, el sprint avanza mediante dos actividades que ocurren en paralelo. La primera es el Desarrollo, que corresponde a la construcción de los ítems comprometidos, ya sean historias de usuario o tareas de modelado, y equivale al trabajo que la Guía Scrum asigna a los Developers dentro de un equipo [13]. La segunda es la Actualización de documentación, que mantiene el Documento de Diseño al día frente a los cambios que va generando el desarrollo. Ambas se detallan en las subsecciones siguientes. En paralelo a ambas, los ayudantes retroalimentan al equipo como parte de su rol de seguimiento. A diferencia de los sprints siguientes, el Sprint 1 todavía no incorpora la Comprobación de seguridad de los datos. En este sprint el equipo construye la base del producto, pero el grueso del manejo de datos todavía no está desarrollado. Funcionalidades como el registro y la gestión de usuarios, el almacenamiento de datos personales, las integraciones con fuentes externas o, en los proyectos con modelo propio, la incorporación de los datos al producto, suelen estar terminados en el Sprint 2. Una revisión en el Sprint 1 cubriría solo una parte menor de los datos que el producto terminará manejando, y habría que repetirla casi por completo en el sprint siguiente. Realizarla desde el Sprint 2, en cambio, permite revisar una

mayor cantidad de datos y de funcionalidades que los tratan, y deja todavía dos sprints para que el equipo corrija lo que se detecte antes de la Feria de Software.

El sprint cierra con la Sprint Review, la Sprint Retrospective y la evaluación de profesores y ayudantes. Con la evaluación de ambos, el Sprint 1 queda aprobado y el equipo puede iniciar el Sprint 2. El proceso de Emprendimiento, si bien corre en paralelo a este sprint, se documenta aparte, en la sección *Emprendimiento* y se puede ver al sección de *Anexos* como queda combinado con el desarrollo del producto.

FORMULARIO DE ENTREGA DEL SPRINT

Objetivo: Registrar al inicio de cada sprint qué se compromete a desarrollar el equipo, para que los profesores puedan contrastarlo con lo planificado en el Documento de Diseño.

Entrada: Documento de Diseño.

Salida: Formulario de entrega del sprint.

Descripción:

El formulario de entrega es el mismo en los tres sprints y para ambas naturalezas de proyecto. Lo que cambia es qué secciones se completan. En los proyectos sin modelo propio, el formulario conserva lo que hoy se entrega al inicio de cada sprint, es decir, las historias de usuario comprometidas con su estimación en puntos de historia [22]. En los proyectos con modelo propio, durante los Sprints 1 y 2 se agrega una sección de tareas de modelado y otra que muestra el reparto de capacidad entre ambos tipos de trabajo. Así, ambos caminos usan el mismo formato y se entregan en la misma fecha, y los profesores revisan un único tipo de documento (ver Tabla 8).

Sección	Contenido	Se completa en
1. Historias de usuario comprometidas	Identificador, historia, criterios de aceptación, puntos de historia y prioridad.	Todos los proyectos.
2. Tareas de modelado comprometidas	Identificador, fase de CRISP-DM, tarea, evidencia que se espera al cierre y puntos de historia.	Proyectos con modelo propio, Sprints 1 y 2, y en el Sprint 3 si quedan ajustes al modelo.
3. Spikes	Pregunta que busca responder, cómo se verificará y puntos de historia.	Ambas naturalezas de proyecto, solo si es necesario y solo en los Sprints 1 y 2.
4. Reparto de capacidad	Puntos de historia totales y porcentaje destinado a historias de usuario y a tareas de modelado.	Proyectos con modelo propio, Sprints 1 y 2.
5. Cambios respecto de lo planificado	Qué cambió respecto de la distribución tentativa del Documento de Diseño y por qué.	Todos los proyectos, si hay cambios.

Tabla 8: Plantilla propuesta para el Formulario de entrega del sprint. Fuente: Elaboración propia.

La sección 5 es la que conecta la entrega con el Documento de Diseño. No busca impedir que el equipo cambie su planificación, ya que el aprendizaje de cada sprint justifica ajustarla, sino que cada cambio quede declarado y explicado en lugar de aparecer solo en el código. En los proyectos con modelo propio, la evidencia esperada de la sección 2 cumple la misma función que los criterios de aceptación de una historia de usuario. Define de antemano qué debe mostrar el equipo para dar por cumplida una tarea de modelado, aunque su resultado no sea el esperado.

Los Spikes de la sección 3 siguen la regla vigente del curso y están disponibles para ambas naturalezas de proyecto. Un proyecto con modelo propio también puede necesitar un Spike que no tenga relación con el modelo, por ejemplo para resolver una duda sobre una integración externa o sobre una tecnología de la interfaz. Ese Spike se trata igual que en cualquier otro equipo, como parte del trabajo que se compromete junto con las historias de usuario, y no como una tarea de modelado.

ACTUALIZACIÓN DE DOCUMENTACIÓN

Objetivo: Mantener el Documento de Diseño al día con las decisiones que el equipo toma durante el desarrollo, para que no queden implícitas en el código.

Entrada: Documento de Diseño vigente y cambios generados por el desarrollo del sprint.

Salida: Documento de Diseño actualizado, con el registro de cambios del sprint.

Descripción:

Esta actividad ocurre en los tres sprints, en paralelo al desarrollo, y debe estar terminada antes de la Sprint Review. En ella el equipo revisa las secciones del Documento de Diseño que se ven afectadas por lo que se construye, que son el diagrama de arquitectura, el modelo de base de datos, los atributos de calidad, los requisitos no funcionales y el uso de IA en el desarrollo. Las historias de usuario no forman parte de esta actividad, porque ya se gestionan en el Formulario de entrega del sprint.

La razón de reservar un momento explícito para esta revisión es la misma que justificó el Documento de Diseño. Sin una instancia dedicada, las decisiones de arquitectura tienden a quedar implícitas dentro del código, en particular cuando buena parte de ese código se genera con asistencia de IA [43]. Un documento que se escribe al cierre del primer semestre y no se vuelve a revisar pierde rápidamente su relación con el producto real, y deja de servir como referencia para el resto del equipo. Por eso la actualización ocurre en cada sprint, junto al desarrollo y no al final del año.

La sección de uso de IA en el desarrollo merece una mención aparte. Durante el semestre es común que el equipo cambie las herramientas o agentes de IA que usa, o el nivel de autonomía con que los usa en cada tipo de tarea. Como se planteó en el marco conceptual, el trabajo dentro del equipo se desplaza desde escribir código hacia dirigir, revisar y corregir lo que genera la IA [42], y ese cambio de rol solo se puede revisar si el equipo deja registro de qué herramientas usa y para qué. Por eso, cuando cambian las herramientas o la forma de usarlas, el equipo actualiza esta sección dentro de la misma actividad, sin que se requiera una actividad aparte.

Para que la actualización no se limite a reemplazar el documento, el equipo deja un registro breve de lo que cambió en el sprint (ver Tabla 9). Este registro es lo que permite a profesores y ayudantes revisar en la Sprint Review qué decisiones se tomaron, sin tener que comparar dos versiones completas del documento.

Sección del documento	Qué cambió	Por qué se decidió	Qué lo originó
Diagrama de arquitectura	Ej. se agrega un servicio de notificaciones.	Ej. la historia requiere avisos en tiempo real y el diseño anterior no lo contemplaba.	Historia de usuario o tarea de modelado.
Uso de IA en el desarrollo	Ej. se cambia la herramienta usada para generar pruebas.	Ej. la anterior generaba pruebas que no cubrían los criterios de aceptación.	Retrospectiva del sprint anterior.
…			

Tabla 9: Plantilla propuesta para el registro de cambios de la Actualización de documentación. Fuente: Elaboración propia.

3.3.2. SPRINT 2: DESARROLLO CON RETROALIMENTACIÓN Y PUNTO DE DECISIÓN

Objetivo: Completar las funcionalidades principales del producto y, en los proyectos con modelo propio, evaluar el modelo frente a su criterio de éxito, para decidir con qué alcance se aborda el Sprint 3.

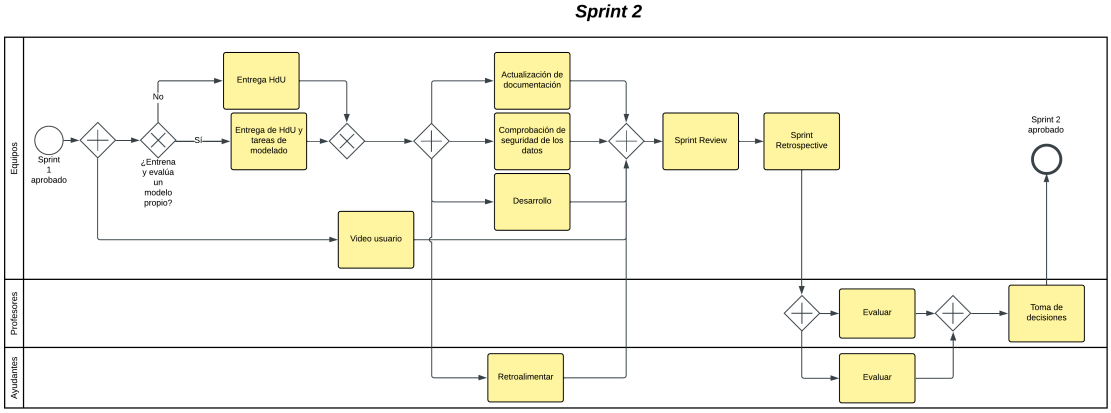


Figura 12: Proceso de Sprint 2. Fuente: Elaboración propia.

Entrada: Sprint 1 aprobado, con su retroalimentación y Documento de Diseño actualizado.

Salida: Incremento del producto presentado en la Sprint Review, Documento de Diseño actualizado, decisión de continuidad del equipo y Sprint 2 aprobado.

Reglas de negocio:

1. El gateway de inicio enruta el flujo según la naturaleza ya clasificada, igual que en el Sprint 1.

2. En los proyectos con modelo propio, la entrega combina historias de usuario con tareas de las fases 5 y 6 de CRISP-DM, con el mismo criterio de reparto del Sprint 1.
3. La comprobación de seguridad de los datos se realiza desde este sprint en adelante, para ambas naturalezas de proyecto.
4. Después de la evaluación del sprint, los profesores deciden si cada equipo continúa en el curso.
5. Los proyectos con modelo propio que no cumplieron su criterio de éxito deben reducir el alcance de su objetivo de Machine Learning para el Sprint 3, sin sacrificar el sprint completo.

Descripción:

El Sprint 2 conserva la misma estructura base que el Sprint 1 (ver Figura 12). El gateway exclusivo según la naturaleza ya clasificada se repite, al igual que la entrega formal con el mismo Formulario de entrega del sprint y las actividades de Desarrollo y Actualización de documentación. Sobre esa base se suman la Comprobación de seguridad de los datos, como una actividad paralela más, y la Toma de decisiones, que ocurre después de la evaluación del sprint. En paralelo a todo el sprint, el equipo realiza además el Video usuario, una instancia que el curso ya realiza, en la que el usuario principal prueba el producto. Durante el sprint los ayudantes retroalimentan al equipo en paralelo a sus actividades, como parte de su rol de seguimiento.

En los proyectos sin modelo propio, la entrega corresponde a las historias de usuario asignadas al Sprint 2 en el Documento de Diseño. En los proyectos con modelo propio, la entrega combina historias de usuario con tareas de las fases 5 (evaluación) y 6 (despliegue) de CRISP-DM, con el mismo criterio de reparto del Sprint 1 y por la misma razón, asegurar que el equipo siempre cuente con un incremento evaluable independiente de cómo avance el trabajo de modelado. La fase de evaluación de CRISP-DM tiene, además, una particularidad que explica por qué el punto de decisión se ubica al cierre de este sprint. A diferencia del ciclo de vida del software tradicional, CRISP-DM contempla explícitamente el retorno a fases anteriores cuando los resultados de la evaluación no son satisfactorios y muestran que los datos o la comprensión del problema deben revisarse [3]. El punto de decisión que cierra este sprint no es un agregado a la propuesta, sino la forma de llevar a Scrum esta no linealidad que CRISP-DM ya reconoce como parte esperada de un proyecto de este tipo.

El sprint cierra con la Sprint Review, la Sprint Retrospective y la evaluación de profesores y ayudantes. A diferencia de los otros sprints, después de la evaluación ocurre la Toma de decisiones, que se detalla en la sección *Toma de decisiones*. Con esta decisión resuelta, el Sprint 2 queda aprobado y el equipo inicia el Sprint 3 con su alcance ya definido.

COMPROBACIÓN DE SEGURIDAD DE LOS DATOS

Objetivo: Verificar que los datos sensibles que maneja el producto se almacenen y traten de forma segura, para que el equipo corrija a tiempo los problemas detectados.

Entrada: Incremento del producto en desarrollo.

Salida: Correcciones aplicadas por el equipo sobre el tratamiento de los datos sensibles.

Descripción:

Esta actividad corre en paralelo al desarrollo desde el Sprint 2 en adelante y es común a ambas naturalezas de proyecto. En ella se verifica que los datos sensibles que maneja el producto (por ejemplo, RUT, contraseñas u otros datos personales) se almacenen y traten de forma segura. Para ello, en cada sprint el equipo revisa lo implementado frente a una lista mínima de puntos. Esta revisión no corresponde a una auditoría técnica del código, sino a comprobar que lo construido cumple con esa lista. La lista mínima considera los siguientes puntos.

- Las contraseñas no se guardan en texto plano.
- Los datos personales solo se solicitan cuando el producto los necesita y se almacenan protegidos.
- El repositorio no contiene credenciales ni claves de acceso.
- En los proyectos con modelo propio, los datos de entrenamiento no exponen datos personales.

Si el equipo detecta un problema, lo corrige durante el mismo sprint. Se sugiere, además, que esta revisión forme parte del seguimiento que los ayudantes ya realizan en cada sprint, contrastando la lista mínima con lo construido por el equipo.

La actividad responde al desajuste de datos personales e inteligencia artificial identificado en la síntesis de la problemática. Con cerca de dos de cada tres proyectos incorporando algún componente de IA y la Ley 21.719, que se espera entre en vigencia en diciembre de 2026, el tratamiento de los datos sensibles no puede seguir quedando implícito en el código [56]. Se incorpora desde el Sprint 2 porque es recién en este sprint cuando se desarrolla el grueso del manejo de datos del producto, como se explicó en el Sprint 1, lo que permite revisar una mayor cantidad de datos en una sola instancia[11].

TOMA DE DECISIONES

Objetivo: Decidir al cierre del Sprint 2 la continuidad de cada equipo y, en los proyectos con modelo propio, el alcance de su objetivo de Machine Learning, para abordar el Sprint 3 con un compromiso alcanzable.

Entrada: Evaluación del Sprint 2 realizada por profesores y ayudantes. En los proyectos con modelo propio, el modelo evaluado en el Sprint 2 y el criterio de éxito registrado en el Documento de Diseño.

[11]Definir los controles de cumplimiento específicos frente a la Ley 21.719, así como automatizar esta revisión mediante scripts, excede el alcance de este trabajo, tal como se planteó en el marco conceptual, y queda planteado como trabajo futuro.

Salida: Decisión de continuidad del equipo en el curso. En los proyectos con modelo propio, el alcance del objetivo de Machine Learning para el Sprint 3, mantenido o reducido.

Descripción:

La Toma de decisiones es la última actividad del Sprint 2 y la única que no se repite en los otros sprints. Está a cargo de los profesores y se organiza en dos decisiones sucesivas. La primera es si el equipo continúa en el curso. Los profesores evalúan el desempeño del equipo a partir de su avance y sus notas durante el semestre, y deciden si sigue adelante. Si no continúa, la actividad termina en ese punto. Esta decisión ya se toma hoy en el curso y no es un aporte de esta propuesta, pero se incluye porque la segunda decisión depende de ella.

La segunda decisión solo aplica a los equipos que continúan y depende de la naturaleza del proyecto. En los proyectos sin modelo propio no hay nada más que decidir, y el equipo pasa directamente al Sprint 3. En los proyectos con modelo propio, los profesores revisan si el modelo evaluado en el Sprint 2 cumple el criterio de éxito definido en el Formulario de Spike durante el Sprint 0 y registrado en el Documento de Diseño, sin modificar ese criterio en esta instancia. Si lo cumple, el equipo inicia el Sprint 3 con el alcance planificado. Si no lo cumple, el equipo reduce el alcance de su objetivo de Machine Learning para el Sprint 3, en lugar de sacrificar el sprint completo, y declara ese cambio en la sección de cambios del Formulario de entrega del Sprint 3.

Reducir el alcance en lugar de sacrificar el sprint completo se apoya en lo planteado en el marco conceptual respecto de qué significa «completado» para un proyecto de este tipo. Penalizar al equipo por una hipótesis de modelado que resultó no viable sería metodológicamente incorrecto, dado que un proyecto de Machine Learning dedica gran parte de su esfuerzo a explorar si los datos disponibles permiten resolver el problema planteado, exploración que no siempre termina en algo entregable [3]. Reducir el alcance reconoce esa incertidumbre como parte esperada del proceso y no como una falla del equipo.

3.3.3. SPRINT 3: CIERRE DEL DESARROLLO

Objetivo: Completar el producto del proyecto, para presentarlo terminado en la Feria de Software.

Sprint 3

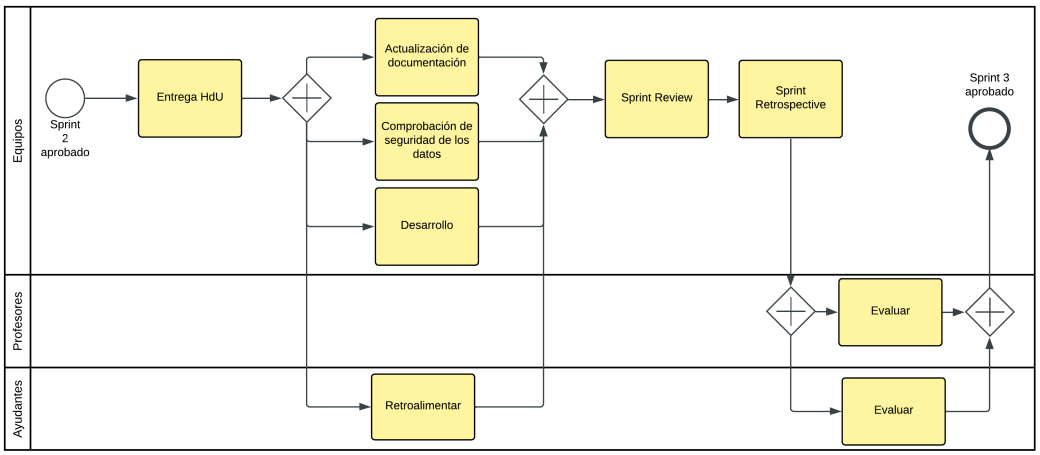


Figura 13: Proceso de Sprint 3. Fuente: Elaboración propia.

Entrada: Sprint 2 aprobado, con su retroalimentación.

Salida: Producto final presentado en la Sprint Review, Documento de Diseño actualizado y Sprint 3 aprobado.

Reglas de negocio:

1. La entrega no distingue entre naturalezas de proyecto y corresponde solo a historias de usuario.
2. En los proyectos con modelo propio, el alcance es el que resultó de la toma de decisiones. No hay tareas de modelado de fondo, pero el equipo puede incluir en la entrega los ajustes al modelo que aún queden por realizar.
3. No se incluyen Spikes, de acuerdo con la regla vigente del curso [22].
4. La Actualización de documentación y la Comprobación de seguridad de los datos continúan igual que en el Sprint 2.

Descripción:

El Sprint 3 conserva la misma estructura base que los sprints anteriores (ver Figura 13), con entrega formal, actividades en paralelo y cierre con Sprint Review, Sprint Retrospective y evaluación. Sin embargo, presenta una diferencia respecto de los dos anteriores, ya que no comienza con un gateway según la naturaleza del proyecto. En este sprint ambos caminos convergen.

Esta convergencia no es una simplificación arbitraria, sino la consecuencia de cómo se distribuyó el trabajo de Machine Learning en los sprints anteriores. Al llegar a este punto, se espera que las seis fases de CRISP-DM hayan sido abordadas al menos una vez, con la comprensión del negocio y de los datos en el Sprint 0, la preparación de los datos y el modelado en el Sprint 1, y la evaluación y el despliegue en el Sprint 2 [3]. Como CRISP-DM es un proceso iterativo, pueden existir iteraciones o ajustes posteriores, pero ya no queda una fase sin abordar que justifique mantener un reparto entre historias de usuario y tareas de

modelado. Esto no significa que el modelo quede cerrado. Siempre pueden quedar ajustes por realizar, en particular cuando la Toma de decisiones redujo el alcance del objetivo de Machine Learning, pero ya no se trata de trabajo de modelado de fondo sino de terminar de afinar lo construido. La entrega del Sprint 3 corresponde, para ambas naturalezas, a las historias de usuario pendientes del Documento de Diseño. En los proyectos con modelo propio, esas historias responden a la integración y el refinamiento del producto en torno al modelo ya evaluado, incluido, cuando corresponde, el alcance reducido que resultó de la Toma de decisiones. Parte de ellas corresponde a funcionalidades deseables u opcionales que permiten al usuario interactuar con lo ya desplegado. Desde el punto de vista de la mecánica del sprint, el equipo trabaja igual que uno sin modelo propio, y usa el mismo Formulario de entrega sin las secciones 3 y 4, completando la sección 2 solo si quedan ajustes al modelo. Esta convergencia es consistente con lo planteado en el marco conceptual, donde la integración de CRISP-DM en la cadencia de Scrum no busca reemplazar el ciclo de vida del software, sino intercalarse con él durante las fases en que corresponde [48].

A partir de la entrega, el sprint avanza mediante tres actividades en paralelo, el Desarrollo, la Actualización de documentación y la Comprobación de seguridad de los datos, descritas en las secciones anteriores, mientras los ayudantes retroalimentan al equipo, igual que en los sprints anteriores. No hay Toma de decisiones, porque el alcance ya quedó resuelto al cierre del Sprint 2. El sprint cierra con la Sprint Review, la Sprint Retrospective y la evaluación de profesores y ayudantes. Con su aprobación termina el desarrollo del producto y el equipo pasa al evento de Feria de Software.

3.3.4. SÍNTESIS COMPARATIVA DE AMBOS CAMINOS

A lo largo de las secciones anteriores, ambos caminos (sin modelo propio y con modelo propio) se describieron por separado en cada etapa. La Tabla 10 reúne esa comparación en un solo lugar, mostrando en qué etapas el contenido efectivamente diverge y en cuáles ya no lo hace.

Etapa	Proyecto sin modelo propio	Proyecto con modelo propio
Sprint 0 y PMV	Historias de usuario críticas, que materializan la propuesta de valor, desarrolladas y presentadas en el PMV.	Spike con las fases 1 y 2 de CRISP-DM (comprensión del negocio y de los datos) más la primera tarea de la fase 3. En el PMV se presenta la evidencia de viabilidad.
Documento de Diseño	Distribución tentativa de historias de usuario a lo largo de los 3 sprints, ordenadas por valor.	Distribución combinada de historias de usuario y tareas de CRISP-DM, con las fases 3 y 4 en el Sprint 1 y las fases 5 y 6 en el Sprint 2. Registra el criterio de éxito del modelo.
Sprint 1	100% historias de usuario.	Reparto acordado con los profesores, con una sugerencia de 60% historias de usuario y 40% tareas de preparación de datos y modelado (fases 3 y 4 de CRISP-DM).
Sprint 2	100% historias de usuario. La Toma de decisiones solo resuelve la continuidad del equipo.	Reparto acordado con los profesores, con una sugerencia de 60% historias de usuario y 40% tareas de evaluación y despliegue (fases 5 y 6 de CRISP-DM). La Toma de decisiones resuelve la continuidad y el cumplimiento del criterio de éxito.
Sprint 3	100% historias de usuario.	Historias de usuario, sin tareas de modelado de fondo, aunque pueden quedar ajustes al modelo según lo resuelto al cierre del Sprint 2.
Actividades paralelas de cada sprint	Desarrollo y Actualización de documentación (desde el Sprint 1) y Comprobación de seguridad de los datos (desde el Sprint 2).	Las mismas, sin diferencias de contenido por naturaleza del proyecto.

Tabla 10: Síntesis comparativa del contenido de cada etapa según la naturaleza del proyecto. Fuente: Elaboración propia.

Leída en conjunto, la tabla muestra que la divergencia entre ambos caminos está concentrada exclusivamente en qué se entrega en cada etapa (el contenido del backlog), nunca en cómo se organizan las ceremonias, los roles o el calendario del semestre, que permanecen idénticos para ambas naturalezas de proyecto en todo momento. Esta concentración de la divergencia es, precisamente, el resultado buscado por la lógica de adecuación, que consiste en conservar la estructura del marco vigente y ajustar solo los puntos donde la diversificación de proyectos genera un desajuste real [1]. Mantener fechas

de entrega, nombres de hito y porcentajes de evaluación para ambos caminos responde además a una razón práctica, evitar que estudiantes, ayudantes y profesores deban manejar dos calendarios o dos criterios de evaluación distintos según la naturaleza del proyecto, lo que sería una fuente adicional de confusión dentro del curso.

La tabla también deja ver que la divergencia no es permanente. Comienza en el Sprint 0 y se extiende hasta el Sprint 2, exactamente mientras queda trabajo de CRISP-DM por realizar, y desaparece por completo en el Sprint 3, cuando se espera que las seis fases hayan sido abordadas al menos una vez. Esta forma de converger hacia el final del proyecto refleja que la adecuación propuesta no crea dos procesos paralelos y permanentemente distintos, sino un único proceso cuyo contenido se ajusta solo mientras la naturaleza del proyecto efectivamente lo requiere.

3.3.5. EVENTO FERIA DE SOFTWARE

Objetivo: Presentar públicamente el producto desarrollado durante el año y su potencial como startup, para someter el proyecto a la evaluación de jueces.

Entrada: Sprint 3 aprobado, con el producto final.

Salida: Evaluación del equipo en el evento, incluida la de su pitch en el Demoday.

Reglas de negocio:

1. El evento se aplica igual a ambas naturalezas de proyecto.
2. Además de la exposición del producto, el equipo presenta su proyecto como startup en un pitch, instancia conocida como Demoday.
3. El pitch del Demoday se presenta ante jueces que pueden ser externos o internos a la asignatura.

Descripción:

Al cierre del Sprint 3, el segundo semestre culmina con el evento de Feria de Software. En la propuesta de Calvo [1], el evento es una jornada de exposición pública de los proyectos en la que se realizan evaluaciones en paralelo. Profesores y ayudantes evalúan el producto y su presentación, mientras que un jurado invitado evalúa el producto y su potencial en la industria a partir de su propuesta de innovación. La evaluación del jurado ya consideraba aspectos cercanos al emprendimiento, como la propuesta de valor, la diferenciación frente al mercado y la proyección del producto, y el proyecto mejor evaluado recibía el Premio al emprendimiento innovador. Cada equipo hoy en día expone su proyecto como una startup mediante un pitch, en una instancia conocida como Demoday, ante jueces que pueden ser externos o internos a la asignatura.

Este cambio es coherente con las mayores exigencias de innovación y emprendimiento que motivan esta propuesta. El pitch del Demoday es, en la práctica, el momento en que el equipo muestra hacia afuera lo que construyó en el proceso completo a lo largo del año, partiendo desde la problemática hasta la propuesta de valor de su solución.

Respecto de la naturaleza del proyecto, el evento no distingue entre caminos. Esto es consistente con lo mostrado en la síntesis comparativa, donde la divergencia entre proyectos sin modelo propio y proyectos con modelo propio desaparece al llegar al Sprint 3. Al momento de la Feria de Software ambos caminos ya convergieron en un único proceso, por lo que no queda ningún desajuste que justifique adaptar esta instancia según la naturaleza del proyecto. Por último, la exposición en la Feria es uno de los momentos de mayor visibilidad del proyecto, y su efecto en los canales del equipo se mide después en el tercer hito de seguimiento del proceso de Emprendimiento.

3.3.6. CIERRE DE PROYECTOS

Objetivo: Registrar las experiencias del equipo durante el año, para que las futuras generaciones las conozcan y eviten repetir los mismos errores.

Entrada: Evaluación obtenida en el evento de Feria de Software y Reporte de Marketing 3.

Salida: Documento de Legado grupal.

Reglas de negocio:

1. El Documento de Legado es grupal y se aplica igual a ambas naturalezas de proyecto.
2. El documento registra qué funcionó y qué no durante el año, los errores cometidos y las recomendaciones para futuros equipos.
3. Además de la introspección que plantea el marco vigente, el documento incluye las lecciones aprendidas sobre los elementos que incorpora esta propuesta.

Descripción:

El cierre de proyectos se fundamenta más en lo que plantea el PMBOK sobre el cierre de un proyecto que en la propuesta de Calvo [1]. Como se definió en el marco conceptual, un proyecto es un esfuerzo temporal con un inicio y un fin definidos [19]. Lo que se aprende durante ese esfuerzo se pierde si nadie lo registra antes de que el equipo se disuelva. Por eso el PMBOK considera el registro de lecciones aprendidas, un documento donde se deja constancia del conocimiento obtenido durante el proyecto para mejorar el desempeño de futuros equipos y de la organización [19]. En la Feria de Software esta situación es aún más marcada, porque cada año los equipos se disuelven al terminar TDPI y la generación siguiente parte desde cero, sin acceso a lo que vivieron quienes ya pasaron por el curso.

El Documento de Legado cumple ese rol dentro del curso. Su objetivo es reunir las experiencias del equipo durante el año, qué funcionó, qué no, qué errores se cometieron y cómo se resolvieron, para que las futuras generaciones las conozcan y no repitan los mismos errores. El documento ya existe en el marco vigente. Calvo [1] lo reformula desde un formato individual, centrado en aprendizajes personales y una evaluación cruzada entre integrantes, hacia un formato grupal estructurado en torno al proyecto, una introspección sobre qué funcionó y qué no, sugerencias para futuros equipos (profesionales, administrativas, técnicas y referencias recomendadas) y un componente de feedback académico sin incidencia en

la nota, dirigido a la coordinación de la asignatura. Esta propuesta retoma la introspección sobre qué funcionó y qué no, pero la entiende desde el PMBOK. El documento no es solo una retroalimentación hacia el curso, sino el registro de lecciones aprendidas que queda disponible para los equipos que vienen.

Para que el documento sea coherente con esta propuesta, además de esa introspección, el equipo registra las lecciones aprendidas sobre los elementos que esta propuesta incorpora al flujo (ver Tabla 11). Así, las futuras generaciones no solo conocen qué funcionó y qué no en términos generales, sino también cómo resultaron en la práctica el Sprint 0 y el PMV, el Documento de Diseño y el uso de IA, el punto de decisión de los proyectos con modelo propio y el Plan de Marketing. En el caso del emprendimiento, los resultados del Reporte de Marketing 3 sirven como insumo, de modo que el equipo deja registro de qué canales funcionaron y cuáles no, y las futuras generaciones cuentan con esa experiencia al construir su propio Plan de Marketing.

Sección	**Qué registra**	**Se construye a partir de**
Introspección del proyecto	Qué funcionó y qué no durante el año, los errores cometidos y cómo se resolvieron.	Experiencia del equipo durante el año.
Sprint 0 y PMV	Si las historias críticas o el Spike permitieron detectar a tiempo los riesgos del proyecto y qué se haría distinto.	Formulario de Historias de Usuario o Formulario de Spike y presentación del PMV.
Documento de Diseño y uso de IA	Qué decisiones de diseño se mantuvieron o cambiaron durante el desarrollo y cómo resultó el uso de IA declarado.	Documento de Diseño actualizado y su registro de cambios.
Modelo propio	Solo en proyectos con modelo propio. Si el modelo cumplió su criterio de éxito y, si se redujo el alcance, por qué.	Formulario de Spike y Toma de decisiones del Sprint 2.
Emprendimiento	Qué canales funcionaron y cuáles no frente a las metas del Plan de Marketing.	Reporte de Marketing 3.

Tabla 11: Plantilla propuesta para el Documento de Legado. Fuente: Elaboración propia a partir de Calvo [1].

3.3.7. EMPRENDIMIENTO

Objetivo: Poner en marcha la marca y el Plan de Marketing del proyecto a partir del modelo de negocios consolidado, para que al terminar la Feria de Software el equipo cuente con una base medible para continuar como emprendimiento.

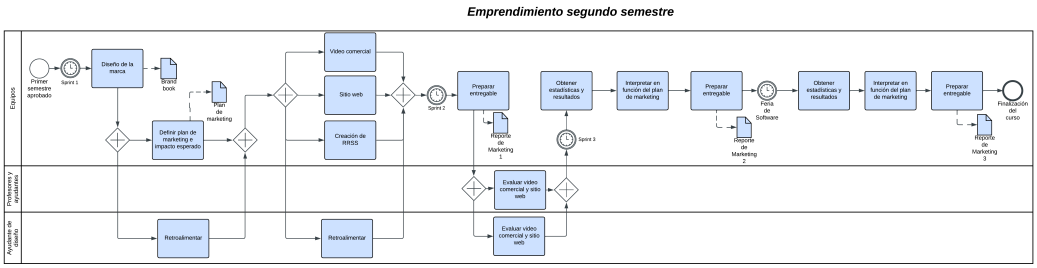


Figura 14: Proceso de emprendimiento del segundo semestre. Fuente: Elaboración propia.

Entrada: Modelo de negocios consolidado, entregado al cierre del primer semestre (ver *Modelo de negocios consolidado*).

Salida: Manual de marca. Plan de Marketing, con la planificación del impacto esperado y los KPI de cada canal. Video comercial, sitio web y redes sociales del proyecto en funcionamiento. Reportes de Marketing 1, 2 y 3, correspondientes a los tres hitos de seguimiento del Plan de Marketing.

Reglas de negocio:

1. El Diseño de la marca y el Plan de Marketing se construyen sobre el modelo de negocios consolidado del primer semestre.
2. La marca, el Plan de Marketing y los canales se desarrollan durante el Sprint 1, de modo que estén en funcionamiento al inicio del Sprint 2.
3. El ayudante de diseño cumple un rol fundamental, ya que retroalimenta las decisiones visuales y de comunicación de la marca (logo, colores, tipografías y textos).
4. La planificación del impacto esperado debe asociar el proyecto a al menos un Objetivo de Desarrollo Sostenible y explicar cómo contribuye a él.
5. Cada canal incluido en el Plan de Marketing debe declarar su rol en el recorrido del cliente (alcance, interés o conversión) y tener al menos un KPI coherente con ese rol, con una meta numérica calculada a partir de la meta de clientes del modelo de negocios consolidado y definida antes de que el canal se ponga en marcha.
6. El proyecto debe contar, como mínimo, con un video comercial, un sitio web y perfiles en Instagram y LinkedIn. Todo el material de estos canales debe respetar el manual de marca.
7. El video comercial y el sitio web se evalúan en conjunto por el ayudante de diseño y por profesores y ayudantes, al cierre del Reporte de Marketing 1.
8. Los Reportes de Marketing 2 y 3 miden los mismos KPI definidos en el Plan de Marketing y se interpretan respecto de sus metas. Un KPI nuevo solo se agrega si el equipo justifica por qué no estaba en el plan.
9. El Reporte de Marketing 3 se prepara después del evento de Feria de Software y debe incluir el efecto de la Feria en los resultados.

Descripción:

Esta propuesta distingue entre innovación y emprendimiento como dos procesos relacionados, pero distintos. La innovación, entendida como la identificación temprana de

problemas, oportunidades y una propuesta de valor diferenciada, ya fue sistematizada por Cayo [2] en las cuatro primeras etapas del primer semestre, y se aborda en el proceso de *Propuesta de innovación y definición del problema*. El trabajo de emprendimiento comienza también en el primer semestre, cuando el equipo consolida su modelo de negocios al cierre del proceso de planificación y diseño.

El emprendimiento como proceso, en cambio, ocurre durante el segundo semestre, en paralelo a los sprints de desarrollo (ver Figura 14 y su versión ampliada en la Figura 22, en los anexos). Para ver en qué momento de cada sprint ocurren las actividades de emprendimiento, los anexos incluyen además una versión combinada de los Sprints 1, 2 y 3 con este proceso (ver Figura 23, Figura 24 y Figura 25). Como se identificó en la síntesis de la problemática del capítulo anterior, hoy no existe una secuencia definida de actividades y entregables que acompañe al equipo en la construcción de su proyecto como posible emprendimiento. El único entregable de este tipo es el Business Model Canvas junto con un modelo de negocio preliminar, que se solicita después del segundo sprint [38]. Lo que ya existe en el marco vigente en esta línea son las actividades de marketing (video, redes sociales, sitio web y un reporte de estadísticas) [1], pero sin una secuencia que conecte el modelo de negocio con lo que se comunica ni con lo que se mide. Esta propuesta organiza esas piezas en un proceso que parte con el Sprint 1 y se extiende hasta después del evento de Feria de Software, con el objetivo de que, al terminar el año, el equipo cuente con una base ordenada para continuar con su proyecto si así lo decide.

DISEÑO DE LA MARCA

Objetivo: Diseñar la identidad de marca del proyecto, para que el equipo la aplique de forma consistente en todos sus canales.

Entrada: Modelo de negocios consolidado.

Salida: Manual de marca.

Descripción:

Al comenzar el Sprint 1, con el negocio ya definido en el modelo de negocios consolidado, el equipo diseña la marca de su proyecto y la registra en un manual de marca, es decir, un documento que reúne la identidad de la marca y las reglas para usarla de forma consistente. Como se definió en el marco conceptual, la gestión de marca no se reduce al logo, abarca la identidad visual, la experiencia que entrega la marca y el lugar que busca ocupar en la mente del cliente [40]. Por eso el manual de marca incluye tanto los elementos de identidad (propósito, misión, valores, personalidad y posicionamiento) como los visuales (logo, paleta cromática, tipografías e iconografía), el tono de comunicación y ejemplos de uso (ver Tabla 12).

En esta etapa participa el o la ayudante de diseño, que retroalimenta al equipo sobre las decisiones visuales y de comunicación de la marca. Su retroalimentación ocurre en paralelo a la definición del Plan de Marketing, de modo que el equipo pueda ajustar la marca antes

de llevarla a sus canales. El marco vigente no contempla este acompañamiento como un rol diferenciado dentro del equipo de ayudantes, aunque se ha usado en las últimas versiones del curso con buenos resultados. La propuesta formaliza esta práctica.

Componente	**Qué debe incluir**
Identidad	Nombre, propósito, misión, valores y personalidad de la marca.
Propuesta y posicionamiento	Propuesta de valor en una frase y el espacio que la marca busca ocupar frente a su competencia.
Identidad visual	Logo y sus variantes (color, monocromo, ícono), paleta cromática con sus códigos, tipografías e iconografía.
Tono de comunicación	Cómo habla la marca, con ejemplos de lo que sí y lo que no diría.
Ejemplos de uso	Aplicación de la marca en al menos una publicación de redes sociales y en la portada del sitio web.

Tabla 12: Plantilla propuesta para el manual de marca. Fuente: Elaboración propia.

PLAN DE MARKETING E IMPACTO ESPERADO

Objetivo: Definir cómo el equipo llega a su segmento y convierte prospectos en clientes, para alcanzar la meta de clientes del modelo de negocios consolidado.

Entrada: Modelo de negocios consolidado, con su meta de clientes, Manual de marca y Lean Canvas del primer semestre.

Salida: Plan de Marketing, con la planificación del impacto esperado y los KPI y metas de cada canal.

Descripción:

Ya iniciado el Sprint 1, el equipo define su Plan de Marketing. Si el modelo de negocios consolidado establece cuántos clientes necesita el proyecto para ser viable, a partir del mercado alcanzable, el Plan de Marketing tiene que responder cómo va a conseguirlos, es decir, cómo el equipo llega a su segmento, despierta su interés y logra que una parte de esas personas u organizaciones pase de ser un prospecto a ser un cliente. Por eso el plan no se plantea como una lista de publicaciones o de redes sociales a crear, sino como una estrategia con una meta concreta que viene del modelo de negocio.

El plan sigue el modelo de diez etapas del material de la asignatura: diagnóstico, investigación, cliente objetivo, propuesta de valor, marca y posicionamiento, objetivos, estrategia digital, implementación, métricas y analíticas, y experimentación [40]. En esta propuesta, esas etapas se agrupan en cinco partes que el entregable debe desarrollar (ver Tabla 13).

Parte del plan	**Etapas del modelo que agrupa**	**Contenido**
Situación de partida	Diagnóstico e investigación	Dónde está el proyecto al iniciar el semestre, qué hace la competencia para llegar al mismo segmento y qué canales usa. Retoma el benchmarking del modelo de negocios consolidado, pero mirado desde la comunicación y no desde el producto.
A quién se dirige	Cliente objetivo y propuesta de valor	Perfil de cliente ideal si el proyecto vende a personas, o perfil de la organización cliente y sus roles de compra (usuario, comprador y decisor) si vende a organizaciones, reutilizando el Template de Cliente Ideal del primer semestre. Este perfil indica también dónde busca información el cliente y qué redes utiliza.
Qué se comunica	Marca y posicionamiento	El mensaje principal y el posicionamiento definidos en el manual de marca, traducidos a lo que se dice en cada canal.
Qué se quiere lograr	Objetivos	La meta de clientes que viene del modelo de negocios consolidado, acotada a lo que es alcanzable dentro del año del curso y traducida en metas para cada canal según el rol que cumple en el recorrido del cliente.
Cómo se logra y cómo se mide	Estrategia digital, implementación, métricas y analíticas, y experimentación	Qué canales y acciones se usan y qué rol cumple cada uno en el recorrido del cliente, un calendario alineado con los sprints, los KPI de cada canal con su meta y qué pruebas hará el equipo para mejorar los resultados que no avancen como se esperaba.

Tabla 13: Partes del Plan de Marketing y su relación con el modelo de diez etapas. Fuente: Elaboración propia a partir del material de la asignatura.

Para definir cómo se llega a la meta de clientes, el plan parte del recorrido que hace una persona desde que conoce el proyecto hasta que se acerca a ser cliente. Ese recorrido tiene tres instantes sucesivos: primero conoce el proyecto (alcance), luego se interesa y busca más información (interés), y finalmente realiza una acción concreta que lo acerca a ser cliente (conversión). En cada instante solo una parte de las personas avanza al siguiente, por lo que la meta de clientes del modelo de negocios consolidado define cuánto interés y cuánto alcance necesita generar el equipo. Por ejemplo, si el plan apunta a 50 clientes y se espera que un 2% de quienes visitan el sitio web se registre, el equipo necesita del orden de 2.500 visitas, y esa cifra a su vez determina cuánto alcance deben lograr sus redes sociales y su video comercial para llevar ese tráfico al sitio.

Los KPI se definen por canal, ya que así es como el equipo efectivamente trabaja su marketing y como cada plataforma entrega sus estadísticas. Para no perder la relación con la meta de clientes, cada canal declara su rol en el recorrido, es decir, en cuál de los tres instantes participa principalmente. No todos los canales cumplen la misma función: las redes sociales y el video comercial suelen servir para dar a conocer el proyecto y llevar personas al sitio web, mientras que el sitio web es donde ocurre la acción concreta. Aun así, el rol no queda fijo para un tipo de canal. En un proyecto que vende a organizaciones, por ejemplo, LinkedIn puede ser el canal donde se concreta el contacto con quien decide la compra y, por lo tanto, cumplir un rol de conversión. El equipo decide el rol de cada canal a partir de su perfil de cliente ideal o perfil de la organización cliente, según dónde busca información ese cliente y en qué parte de su recorrido aparece cada canal.

Declarar el rol de cada canal cumple tres funciones. Primero, orienta qué KPI tiene sentido medir. A un canal cuyo rol es dar a conocer el proyecto se le piden KPI de alcance, como seguidores o reproducciones, y no tiene sentido exigirle conversiones, mientras que al sitio web se le pide su tasa de conversión. Segundo, ordena el cálculo de las metas. La meta del canal de conversión sale directamente de la meta de clientes, y las metas de los canales de alcance e interés se calculan a partir del tráfico que ese canal necesita recibir. Así, las metas no se fijan de forma arbitraria, sino que se calculan desde la meta de negocio. Tercero, permite que en los hitos de seguimiento el equipo identifique en qué parte del recorrido está perdiendo a sus potenciales clientes. Si un canal de alcance no cumple su meta, el problema está en dar a conocer el proyecto; si el alcance se cumple pero el sitio no convierte, el problema está en lo que el sitio ofrece o comunica.

Dentro del año del curso, la conversión no siempre corresponde a una venta. Según la naturaleza del proyecto, puede ser un registro, una inscripción en lista de espera, una solicitud de demostración o, en proyectos que venden a organizaciones, una reunión con quien decide la compra. Lo importante es que el equipo defina cuál es esa acción, que la mida y que entienda qué la separa de convertirse en un cliente.

En esta misma actividad se planifica el impacto esperado. El equipo justifica a qué Objetivo de Desarrollo Sostenible contribuye su proyecto y cómo espera lograrlo [9]. Esto, junto con las metas de cada canal, no estaba contemplado en el marco vigente, donde los canales se construían sin metas definidas de antemano [1]. Definir las metas aquí, antes de poner en marcha los canales, es lo que permite que los hitos de seguimiento comparen lo logrado con lo planificado, en vez de describir los resultados después de obtenerlos. La Tabla 14 propone el formato de esta definición, que luego se reutiliza en los Reportes de Marketing 2 y 3.

Canal	Rol en el recorrido	KPI	Meta
Instagram	Alcance: dar a conocer el proyecto al segmento y llevar personas al sitio web.	Seguidores, alcance promedio, tasa de interacción (engagement).	Calculada desde el tráfico que necesita el sitio web.
LinkedIn	Alcance o interés. Conversión, si el proyecto vende a organizaciones y el contacto con quien decide la compra ocurre aquí.	Seguidores, tasa de interacción; contactos concretados, si su rol es de conversión.	Según su rol.
Video comercial	Alcance e interés: presentar el problema y la solución, e invitar a visitar el sitio.	Reproducciones, clics hacia el sitio web.	Calculada desde el tráfico que necesita el sitio web.
Sitio web	Conversión: llevar al visitante a una acción concreta (registro, lista de espera, demo o contacto).	Visitas, tasa de clics (CTR), tasa de conversión.	Meta de clientes del modelo de negocios consolidado, acotada al año del curso.

Tabla 14: Plantilla propuesta para la definición de KPI y metas por canal del Plan de Marketing. Fuente: Elaboración propia.

PUESTA EN MARCHA DE LOS CANALES

Objetivo: Construir los canales definidos en el Plan de Marketing, para comenzar a llegar al segmento de clientes de forma coherente con lo planificado y con la identidad de la marca.

Entrada: Plan de Marketing y Manual de marca.

Salida: Video comercial, sitio web y redes sociales del proyecto en funcionamiento.

Descripción:

Con el Plan de Marketing definido, el equipo construye en paralelo sus tres canales: el video comercial, el sitio web y las redes sociales del proyecto. Las tres actividades ya estaban presentes en el marco vigente, que las repartía entre los sprints, con el sitio web y las redes sociales en el Sprint 1 y el video comercial en el Sprint 2 [1]. La diferencia es que ahora cada canal responde a lo definido en el plan y aplica la identidad del manual de marca. Mientras el equipo trabaja en ellos, el ayudante de diseño vuelve a retroalimentar sobre su componente visual.

Para cada canal se define qué se espera de él. El sitio web debe comunicar la propuesta de valor, orientar al usuario hacia una acción concreta y ser accesible. El video comercial debe presentar el problema, mostrar la solución en funcionamiento y transmitir un mensaje claro. Las redes sociales deben publicar contenido consistente con el tono definido en el manual de marca y dirigido al segmento del perfil de cliente ideal.

HITO DE SEGUIMIENTO 1: REPORTE DE MARKETING 1

Objetivo: Revisar la puesta en marcha del Plan de Marketing, para establecer los valores iniciales de los KPI de cada canal.

Entrada: Plan de Marketing y Canales en funcionamiento.

Salida: Reporte de Marketing 1.

Descripción:

El Plan de Marketing no se entrega una vez y se da por cerrado, sino que se sigue durante el resto del semestre en tres hitos de seguimiento, cada uno con su propio Reporte de Marketing: el primero en el Sprint 2, el segundo en el Sprint 3 y el tercero después del evento de Feria de Software. Cada hito responde una pregunta distinta sobre el plan, el primero revisa si lo planificado se puso en marcha, el segundo si los canales avanzan hacia las metas definidas y el tercero cuánto se logró al cierre del año. En el marco vigente, los resultados del marketing se revisaban en una única instancia, el reporte de marketing del Sprint 3 [1]. En la práctica actual del curso se piden tres reportes de marketing, pero repartidos de otra forma y sin un Plan de Marketing que los conecte. Organizarlo en tres hitos sobre un mismo plan es lo que permite al equipo ver la evolución de sus canales y corregir a tiempo lo que no funciona.

En el primer hito, el equipo presenta en el Reporte de Marketing 1 cómo implementó su Plan de Marketing: los canales en funcionamiento, el rol que cumple cada uno en el recorrido del cliente y las acciones realizadas respecto de las planificadas en el calendario del plan. Como los canales llevan poco tiempo funcionando, este hito todavía no evalúa si se cumplen las metas, sino que registra los valores iniciales de los KPI de cada canal. Esos valores son el punto de partida contra el que se comparan los dos hitos siguientes (ver Tabla 15).

Al entregarlo, el video comercial y el sitio web se evalúan en paralelo por el ayudante de diseño y por profesores y ayudantes. Son los dos productos del proceso con mayor componente visual, por lo que su evaluación combina la mirada de diseño con la del curso.

Canal	KPI	Meta	Valor anterior	Valor actual	Interpretación y ajustes
Instagram	Tasa de interacción	Meta del plan	Valor del RM anterior	Valor medido	¿Se cumple la meta? ¿Por qué? ¿Qué se ajusta?
Sitio web					
…					

Tabla 15: Plantilla propuesta para los Reportes de Marketing 1, 2 y 3. Fuente: Elaboración propia.

HITO DE SEGUIMIENTO 2: REPORTE DE MARKETING 2

Objetivo: Revisar el avance de los canales hacia las metas del Plan de Marketing, para identificar qué debe ajustar el equipo.

Entrada: Reporte de Marketing 1 y estadísticas de cada canal.

Salida: Reporte de Marketing 2.

Descripción:

En el segundo hito, durante el Sprint 3 y con los canales ya funcionando durante un sprint completo, el equipo obtiene las estadísticas de cada canal y las interpreta en función del Plan de Marketing, es decir, revisa si está obteniendo lo que se propuso en la planificación del impacto esperado y, si no, por qué y qué va a ajustar. Esta instancia es la evolución del Reporte de Marketing Digital de Calvo [1], cuya versión revisada ya avanzaba hacia una tabla estructurada con estadísticas y análisis por canal. Esta propuesta retoma esa tabla y la ancla a las metas definidas en el plan (ver Tabla 15).

HITO DE SEGUIMIENTO 3: REPORTE DE MARKETING 3

Objetivo: Medir el resultado de los canales al cierre del año, incluyendo el efecto de la exposición en la Feria de Software, para que el equipo cuente con una base para continuar su emprendimiento.

Entrada: Reporte de Marketing 2 y resultados del evento de Feria de Software.

Salida: Reporte de Marketing 3, insumo para el Documento de Legado del cierre de proyectos.

Descripción:

En el tercer hito, después del evento de Feria de Software, el equipo repite la medición e interpretación con los mismos KPI y prepara el último reporte. Este muestra la evolución respecto de los dos anteriores e incluye el efecto de la exposición en la Feria. En el marco vigente no existe una instancia de marketing posterior al evento [1], por lo que el

seguimiento del plan en tres hitos es un aporte de esta propuesta. Sus resultados son además insumo para el Documento de Legado del cierre de proyectos.

Con este proceso, el equipo termina el año con un manual de marca, un Plan de Marketing y la evidencia de sus tres hitos de seguimiento sobre el desempeño de los canales, todo construido sobre el modelo de negocios consolidado del primer semestre. Es precisamente la base que, según la síntesis de la problemática, hoy no está definida como un proceso dentro del curso y que permite al equipo continuar con su proyecto como emprendimiento después de la Feria de Software.

3.4. SÍNTESIS DE LA PROPUESTA

A lo largo de este capítulo se desarrolló una propuesta de adecuación del flujo de procesos de la Feria de Software, organizada en procesos por semestre y en torno a las tres condiciones planteadas en el Enfoque de la propuesta, que son las mayores exigencias curriculares en innovación y emprendimiento, la diversificación de los proyectos hacia Machine Learning y el rol de la inteligencia artificial como herramienta de desarrollo. La Tabla 16 resume qué se mantuvo sin cambios respecto del marco vigente de Calvo [1] y qué se adaptó como resultado del diagnóstico presentado en la *Síntesis de la problemática* del capítulo anterior. Se agrega además una fila con los ajustes que no dependen de una condición en particular y que aplican a todos los proyectos.

Condición	Se mantiene	Se adapta
Innovación y emprendimiento	Etapas de innovación definidas por Cayo [2]; cadencia de los sprints y del evento de Feria de Software, sin ceremonias adicionales para el proceso de emprendimiento.	Registro de las etapas de innovación con las plantillas de Lean Canvas del 3IE; modelo de negocios consolidado al cierre del primer semestre, con Business Model Canvas completo y meta de clientes; proceso de Emprendimiento en el segundo semestre con Diseño de marca, Plan de Marketing y tres hitos de seguimiento (Reportes de Marketing 1 a 3); incorporación del o la ayudante de diseño.
Diversificación hacia Machine Learning	Ceremonias, roles y calendario de los Sprints 1, 2 y 3; decisión de continuidad del equipo al cierre del Sprint 2.	Etapa de clasificación de naturaleza de proyecto; Sprint 0 mediante un Spike y su formulario; contenido del PMV y del Documento de Diseño según la naturaleza; reparto acordado de historias de usuario y tareas CRISP-DM en los Sprints 1 y 2, con una sugerencia de 60/40; reducción del alcance del objetivo de Machine Learning al cierre del Sprint 2 si no se cumple el criterio de éxito.
Rol de la inteligencia artificial como herramienta de desarrollo	Estructura general de la planificación de fin de primer semestre.	Renombre y ampliación del Plan de Proyecto a Documento de Diseño (arquitectura, modelo de base de datos, atributos de calidad, uso de IA en el desarrollo); actividad de Actualización de documentación en los tres sprints; actividad de Comprobación de seguridad de los datos desde el Sprint 2.
Común a todos los proyectos	Video usuario en el Sprint 2; evento de Feria de Software; estructura del Documento de Legado.	PMV que debe demostrar la propuesta de valor del proyecto; formulario de entrega común a los tres sprints; entrega del Sprint 1 centrada en las historias esenciales e importantes; documentación del Demoday dentro de la Feria; Cierre de proyectos fundamentado en el registro de lecciones aprendidas del PMBOK, con lecciones sobre los elementos de la propuesta en el Documento de Legado.

Tabla 16: Síntesis de las condiciones que motivan la propuesta, contrastando lo que se mantiene del marco vigente con lo que se adapta en esta propuesta. Fuente: Elaboración propia.

Leída en conjunto con la Tabla 10 presentada en la sección anterior, esta síntesis muestra que el principio de adecuación planteado al inicio del capítulo se sostuvo a lo largo de las tres condiciones abordadas. La propuesta conserva las ceremonias de Scrum, los roles de profesores, ayudantes y equipos y el calendario general definidos por el marco vigente, y concentra la mayor parte de los ajustes en el contenido de los artefactos que los equipos ya entregan hoy, como el backlog, el prototipo inicial, que pasa a ser el PMV, el Plan de Proyecto, que pasa a ser el Documento de Diseño, y las actividades de marketing de cada sprint, que se reorganizan en un Plan de Marketing con tres Reportes de Marketing. Los elementos que sí se agregan al flujo, que son la etapa de clasificación, las actividades paralelas de los sprints, el proceso de Emprendimiento y la formalización del rol del o la ayudante de diseño, son acotados y responden cada uno a un desajuste concreto identificado en la síntesis de la problemática del capítulo 2, sin reemplazar ninguna etapa existente. Las instancias que ya se realizan hoy en el curso, como el Video usuario, la decisión de continuidad del equipo y el Demoday, se documentan en el flujo para que este represente el proceso real, aclarando en cada caso que no son un aporte de esta propuesta.

Este capítulo da cumplimiento al cuarto objetivo específico. Los tres primeros objetivos específicos (el diagnóstico del marco de 2019, el análisis de cómo la diversificación de los proyectos y el uso de IA como herramienta de desarrollo han cambiado la realidad del curso, y la identificación de los marcos ágiles vigentes en la industria y su contraste con el marco de referencia actual) se desarrollaron en los capítulos anteriores y alimentaron directamente las decisiones de diseño documentadas en este capítulo.

Consistente con el alcance definido, esta propuesta no incluye el desarrollo de una herramienta de software que dé soporte o automatice los procesos aquí descritos, ni su implementación como piloto para medir resultados en un contexto real. Ambas quedan planteadas como trabajo futuro, al igual que la definición de los controles de cumplimiento frente a la Ley 21.719. Lo que sí corresponde dentro del alcance de este trabajo es la validación de la propuesta mediante una revisión de su componente de innovación y emprendimiento con un especialista del 3IE y una encuesta aplicada a profesores, ayudantes y estudiantes de GPI y TDPI (el quinto y último objetivo específico), que se presenta en el capítulo siguiente.

CAPÍTULO 4
VALIDACIÓN DE LA SOLUCIÓN

Este capítulo da cumplimiento al quinto objetivo específico: validar la propuesta mediante una retroalimentación con los actores de GPI y TDPI. Como se estableció en el alcance, la propuesta no se implementa como piloto durante el desarrollo de este trabajo, por lo que su validación no mide resultados en un semestre real, sino el juicio que expresan sobre ella quienes conocen el curso o el ámbito que la propuesta aborda.

La validación se realizó en dos instancias complementarias. La primera fue una revisión cualitativa con el Instituto Internacional para la Innovación Empresarial (3IE) de la UTFSM [59], centrada en la estructura del primer semestre y en el proceso de emprendimiento del segundo. La segunda fue una encuesta aplicada a profesores, ayudantes y estudiantes de GPI y TDPI, que abarca el conjunto de los ajustes propuestos. La primera instancia aporta la mirada de un especialista en innovación y emprendimiento sobre la parte de la propuesta más alejada de la ingeniería de software, y la segunda recoge la percepción de quienes ejecutarían el flujo en la práctica.

4.1. DISEÑO DE LA VALIDACIÓN

La revisión con el 3IE sigue la lógica del juicio de expertos, entendido como la opinión informada de personas con trayectoria en el tema, reconocidas como expertos cualificados en él, que pueden entregar información, evidencia, juicios y valoraciones [60]. En este trabajo no se aplicó un procedimiento formal de juicio de expertos con un panel y un instrumento de puntuación, sino una revisión cualitativa, en la que se presentó la propuesta y se recogieron las observaciones del especialista. Sus observaciones se incorporaron a la propuesta antes de aplicar la encuesta, de modo que esta evaluara la versión ya ajustada.

La encuesta, en cambio, busca una valoración cuantitativa de cada ajuste por parte de los tres actores del curso, complementada con preguntas abiertas.

4.2. VALIDACIÓN CON EL INSTITUTO INTERNACIONAL PARA LA INNOVACIÓN EMPRESARIAL

4.2.1. OBJETIVO

Evaluar, desde la experiencia del 3IE en el acompañamiento de emprendimientos de base tecnológica, si la secuencia de innovación y emprendimiento propuesta es coherente con la forma en que se desarrolla un emprendimiento en etapa temprana y si su carga es abordable para equipos que, en paralelo, desarrollan su producto.

4.2.2. PROCEDIMIENTO

La revisión se realizó el 8 de octubre de 2026 con Juan Vergara, subgerente de Emprendimiento del 3IE. Se expuso una presentación de la propuesta que recorrió el problema y la comparación entre el flujo vigente y el propuesto. Luego se detallaron los procesos del primer semestre (Propuesta de innovación y definición del problema, Definición del alcance y riesgos, y Planificación y diseño) y el proceso de emprendimiento del segundo semestre (Diseño de la marca, Plan de Marketing e impacto esperado, Puesta en marcha de los canales e Hitos de seguimiento). A medida que se realizaba la exposición se abría una conversación en la que el especialista comentaba cada parte.

La versión presentada difería de la actual en dos puntos que son, justamente, los que se ajustaron a partir de esta revisión: el entregable de negocio se llamaba Plan de negocios y se desarrollaba al inicio del segundo semestre, y el Plan de Marketing incluía un mapa del recorrido del cliente.

4.2.3. RESULTADOS

Las observaciones recogidas se agrupan en cuatro temas.

Valoración general de la estructura. La propuesta fue valorada positivamente como un todo y se destacó que se encuentra bien estructurada, en especial el primer semestre. Esta parte es la que integra el marco de innovación de Cayo [2] con las plantillas del Lean Canvas del propio 3IE, y la que cierra con la planificación del negocio y del producto antes del segundo semestre.

Carga del segundo semestre y mapa del recorrido del cliente. El 3IE consideró que el proceso de emprendimiento del segundo semestre concentraba una carga alta, en particular por el mapa del recorrido del cliente. En la versión presentada, este mapa formaba parte de la sección del Plan de Marketing dedicada a describir a quién se dirige el proyecto. A partir del perfil de cliente ideal, el equipo debía construir un mapa del recorrido completo de su cliente, desde que descubre una necesidad hasta que decide, y usarlo para definir el rol de cada canal en ese recorrido. Esto implicaba que, durante el Sprint 1 y en paralelo al desarrollo del producto, el equipo debía elaborar el manual de marca, el perfil de cliente ideal, el mapa del recorrido del cliente y el Plan de Marketing, y además construir sus canales. Como se describió en el marco conceptual, el mapa del recorrido del cliente representa todas las etapas, interacciones y experiencias del cliente [40], por lo que su elaboración agrega un entregable completo a una etapa ya exigida. El especialista precisó que el mapa del recorrido del cliente es una herramienta muy valiosa, pero compleja de enseñar y de aplicar correctamente, y que en un periodo tan acotado en el semestre los equipos difícilmente alcanzarían a comprenderla y aplicarla del todo.

Nombre y ubicación del Plan de negocios. Se observó que el nombre Plan de negocios aludía a un documento más extenso del que efectivamente se pedía, ya que históricamente los planes de negocios son documentos largos y detallados. Se recomendó cambiar su nombre a modelo de negocios consolidado y adelantarlo al primer semestre, principalmente por la carga académica del segundo.

Uso del perfil de cliente ideal. Se valoró positivamente el uso del perfil de cliente ideal como herramienta para describir al cliente objetivo y orientar a partir de él las decisiones del Plan de Marketing.

4.2.4. AJUSTES INCORPORADOS

La Tabla 17 resume cómo se incorporaron estas observaciones en la propuesta presentada en el capítulo anterior.

Observación	Ajuste en la propuesta	Sección afectada
Estructura del primer semestre bien armada.	Se mantiene la secuencia de tres procesos del primer semestre.	Estructura primer semestre
Carga alta en el segundo semestre, en especial por el mapa del recorrido del cliente.	Se elimina el mapa del recorrido del cliente del Plan de Marketing. El rol de cada canal se define a partir del perfil de cliente ideal o del perfil de la organización cliente, según dónde busca información el cliente y qué redes utiliza.	Plan de Marketing e impacto esperado
El nombre Plan de negocios sugiere un documento más extenso.	El entregable pasa a llamarse modelo de negocios consolidado y se organiza en cuatro secciones acotadas, que retoman las plantillas del Lean Canvas 3IE.	Modelo de negocios consolidado
Conviene adelantar el entregable por carga académica.	El modelo de negocios consolidado se entrega al cierre del primer semestre, dentro del proceso de Planificación y diseño, y pasa a ser la entrada del proceso de emprendimiento del segundo semestre.	Planificación y diseño, y Emprendimiento
Buen uso del perfil de cliente ideal.	Se mantiene como base del Plan de Marketing.	Plan de Marketing e impacto esperado

Tabla 17: Observaciones del 3IE y ajustes incorporados en la propuesta. Fuente: Elaboración propia.

4.3. ENCUESTA A PROFESORES, AYUDANTES Y ESTUDIANTES

4.3.1. OBJETIVOS DE LA ENCUESTA

Objetivo general: validar, desde la percepción de profesores, ayudantes y estudiantes de GPI y TDPI, si los ajustes propuestos al flujo de procesos de la Feria de Software son adecuados, útiles y factibles, y confirmar que estos actores reconocen los desajustes que les dieron origen.

Objetivos específicos:

1. Confirmar que los actores perciben los desajustes identificados en el diagnóstico: la diversificación de los proyectos hacia Machine Learning, el uso de la IA como herramienta de desarrollo y la falta de un proceso definido de emprendimiento.
2. Evaluar si cada uno de los ajustes propuestos responde al problema que lo motiva y si resulta útil.
3. Evaluar la factibilidad de incorporar los ajustes dentro de los tiempos y créditos de las asignaturas.
4. Comparar la percepción de la propuesta entre profesores, ayudantes y estudiantes, y entre estudiantes de proyectos de distinta naturaleza.
5. Recoger sugerencias de mejora y aspectos difíciles de aplicar.

El primer objetivo específico de la encuesta permite, además, contrastar con la percepción de profesores, ayudantes y estudiantes el diagnóstico presentado en la síntesis de la problemática del capítulo 2.

4.3.2. DISEÑO DEL INSTRUMENTO

La encuesta se aplicó mediante un formulario en línea y anónimo, dirigido a los profesores, ayudantes y estudiantes que participan de GPI y TDPI en el segundo semestre de 2026, en los campus Casa Central y San Joaquín. Se organiza en cuatro partes:

- **Perfil:** campus, rol en la Feria de Software y, solo para estudiantes, la naturaleza de su proyecto (software tradicional, software que integra una IA ya entrenada o proyecto que entrena su propio modelo).
- **Diagnóstico:** afirmaciones sobre el estado actual del curso, presentadas antes de mostrar la propuesta para no condicionar las respuestas.
- **Propuestas:** una sección por cada ajuste, con una breve descripción de qué se propone, el diagrama del flujo correspondiente y una frase que indica qué se busca saber en esa sección.
- **Valoración general:** afirmaciones sobre la propuesta completa y dos preguntas abiertas, una sobre cambios o mejoras y otra sobre lo más difícil de aplicar.

Cada afirmación se responde en una escala de acuerdo de cinco niveles (muy en desacuerdo, en desacuerdo, ni de acuerdo ni en desacuerdo, de acuerdo y muy de acuerdo), a la que se agrega la opción «No sé». Esta opción es necesaria porque todos los participantes recorren todas las secciones, y no todos tienen experiencia con cada parte del curso, por ejemplo, un estudiante de un proyecto de software tradicional respecto del camino de Machine Learning.

Cada afirmación mide una de cinco dimensiones: diagnóstico (si el problema existe), coherencia (si el ajuste responde al problema que lo motiva), utilidad (si aporta al trabajo del equipo), claridad (si se entiende cómo funciona) y factibilidad (si cabe en los tiempos y créditos del curso). La Tabla 18 resume el objetivo de medición de cada sección.

Sección	**Objetivo de medición**	**Afirmaciones**
Diagnóstico	Confirmar que los actores perciben los desajustes en los tipos de proyecto, el uso de IA y el emprendimiento.	3
Clasificación por naturaleza del proyecto	Determinar si la clasificación orienta el desarrollo y si su criterio se entiende.	2
Sprint 0 y presentación del PMV	Determinar si la práctica actual funciona como control de riesgo y si conviene formalizarla.	2
Camino para proyectos de Machine Learning	Confirmar el problema de los proyectos de ML y evaluar el Spike, el PMV, los sprints combinados, el punto de decisión y la factibilidad del Spike.	7
Modelo de negocios consolidado	Determinar si adelantar el modelo de negocios es útil y factible en el primer semestre.	2
Documento de Diseño y uso de IA	Determinar si justificar el diseño y declarar el uso de IA ayuda a que las decisiones no queden implícitas en el código, y si es factible completarlo en el primer semestre.	3
Comprobación de seguridad de datos	Determinar si la comprobación es necesaria y factible dentro del sprint.	2
Emprendimiento en el segundo semestre	Confirmar el problema actual del marketing y evaluar el manual de marca, el Plan de Marketing, las metas por canal, los hitos de seguimiento, su factibilidad y el aporte del ayudante de diseño.	7
Valoración general	Medir la valoración global y la disposición a aplicar la propuesta.	2

Tabla 18: Secciones de la encuesta y su objetivo de medición. Fuente: Elaboración propia.

4.3.3. ANÁLISIS DE LOS DATOS

Las respuestas en escala de acuerdo son datos ordinales: sus niveles tienen un orden, pero la distancia entre ellos no es necesariamente la misma, por lo que no corresponde describirlos con promedios [61]. Por esta razón, cada afirmación se resume con tres indicadores: el porcentaje de acuerdo (respuestas «de acuerdo» y «muy de acuerdo» sobre el total, sin contar «No sé»), la mediana y el porcentaje de respuestas «No sé». En las afirmaciones de diagnóstico redactadas en sentido positivo, como «el flujo de procesos actual se ajusta a los proyectos», se considera el porcentaje de desacuerdo, ya que es este el que confirma el problema. Los resultados se analizan para el total de participantes y por rol, y las respuestas de estudiantes se separan además según la naturaleza de su proyecto.

Para decidir si un ajuste queda validado se fijó, antes de revisar las respuestas, el criterio de la Tabla 19. Las respuestas abiertas se agrupan por tema y se usan para explicar los resultados de las afirmaciones y orientar los ajustes.

Resultado	**Criterio**	**Consecuencia**
Validado	Todas las afirmaciones de la sección alcanzan al menos 70% de acuerdo.	El ajuste se mantiene.
Validado con observaciones	Alguna afirmación alcanza entre 50% y 69% de acuerdo.	El ajuste se mantiene y se revisa el aspecto con menor acuerdo a partir de las respuestas abiertas.
No validado	Alguna afirmación queda bajo 50% de acuerdo.	El ajuste se revisa o queda planteado como trabajo futuro.

Tabla 19: Criterio de validación de cada ajuste. Fuente: Elaboración propia.

4.3.4. RESULTADOS

DESCRIPCIÓN DE LA MUESTRA

La encuesta estuvo abierta entre el 9 y el 13 de octubre de 2026 y reunió 63 respuestas, de las cuales 51 corresponden a estudiantes, 8 a ayudantes y 4 a profesores (ver Tabla 20). Ambos campus quedaron representados, con 35 respuestas de Casa Central y 28 de San Joaquín. Entre los estudiantes predominan los proyectos con modelo propio, que suman 31, seguidos por los proyectos que integran una IA ya entrenada, con 15, y los de software sin IA, con 5. Esta distribución es consistente con el levantamiento de proyectos presentado en el capítulo 2, donde la mayoría de los proyectos recientes incorpora algún componente de IA.

Rol	**Casa Central**	**San Joaquín**	**Total**
Estudiantes	29	22	51
Proyecto con modelo propio			31
Proyecto que integra una IA ya entrenada			15
Software sin IA			5
Ayudantes	4	4	8
Profesores	2	2	4
Total	**35**	**28**	**63**

Tabla 20: Participantes de la encuesta según rol, campus y naturaleza del proyecto. Fuente: Elaboración propia.

Como el número de ayudantes, profesores y estudiantes de software sin IA es acotado, los resultados de esos grupos se presentan como una tendencia y no se usan por sí solos para decidir si un ajuste queda validado. La decisión se toma sobre el total de respuestas, según el criterio de la Tabla 19.

DIAGNÓSTICO

La Figura 15 muestra las respuestas a las afirmaciones sobre el estado actual del curso, que se presentaron antes de mostrar la propuesta. Tres de los desajustes identificados en la síntesis de la problemática se confirman con claridad. El 98 % está de acuerdo con que el uso de herramientas de IA cambió la forma en que los equipos desarrollan sus proyectos, el 81 % con que las actividades de marketing se trabajan hoy sin una conexión clara con el modelo de negocio, y el 71 % con que los equipos con proyectos de ML suelen priorizar las mejoras visibles del producto por sobre la calidad del modelo o de los datos. En esta última afirmación, 12 participantes marcaron «No sé», lo que corresponde al 19 % y se mantiene bajo el límite de 30 % definido para considerarla evaluable.

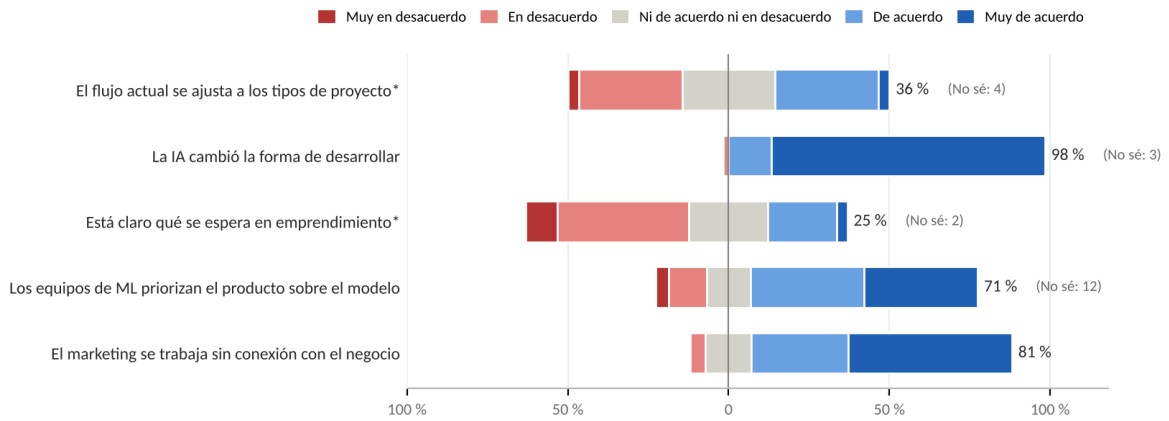


Figura 15: Respuestas a las afirmaciones de diagnóstico. El porcentaje indica las respuestas de acuerdo y muy de acuerdo sin contar «No sé». En las afirmaciones marcadas con asterisco, el desacuerdo es el que confirma el problema. Fuente: Elaboración propia.

Las dos afirmaciones generales, en cambio, no alcanzan el umbral. Ante la afirmación de que el flujo actual se ajusta a los tipos de proyecto que llegan hoy a la Feria, el 36 % está en desacuerdo y otro 36 % de acuerdo, por lo que el problema no se confirma. Ante la afirmación de que hoy está claro qué se espera en la parte de emprendimiento, el 51 % está en desacuerdo, lo que confirma el problema solo de forma parcial. En ambos casos la diferencia entre roles es marcada. Entre los estudiantes, el 43 % considera que el flujo no se ajusta y el 59 % que lo esperado en emprendimiento no está claro, mientras que entre los ayudantes solo uno de ocho comparte cada una de esas opiniones y entre los profesores predominan las respuestas neutrales.

Esto permite contrastar el diagnóstico del capítulo 2. Los desajustes concretos, como el efecto de la IA, la desconexión del marketing y el trabajo de los proyectos de ML, son reconocidos por los tres actores. La percepción general de que el flujo no se ajusta, en cambio, se concentra en los estudiantes, que son quienes lo ejecutan, mientras que ayudantes y profesores, que lo acompañan y lo evalúan, lo perciben como más adecuado. Una lectura posible es que la afirmación se refiere al flujo de procesos de referencia, definido por Calvo [1] en 2019, pero cada participante la responde desde el curso que conoce hoy. Desde entonces, el curso ha incorporado varias prácticas que no están en ese marco, como pedir que el Sprint 0 aborde las historias críticas, contar con un ayudante de diseño o solicitar el Business Model Canvas después del segundo sprint. Ayudantes y profesores, que conocen y aplican esas prácticas, pueden considerar que el flujo actual sí se ajusta porque lo evalúan junto con esas soluciones informales. Los estudiantes, en cambio, viven el flujo desde la ejecución, donde esas prácticas no siempre están explicadas ni conectadas entre sí, y por eso el desajuste les resulta más visible. Esto no contradice el diagnóstico del capítulo 2, sino que muestra que parte del problema ya se está resolviendo en la práctica sin quedar documentada en el flujo, que es precisamente lo que esta propuesta busca formalizar.

RESULTADOS POR PROPUESTA

La Figura 16 muestra la distribución de respuestas de cada afirmación de las siete propuestas y de la valoración general, y la Tabla 21 el porcentaje de acuerdo para el total y para cada rol.

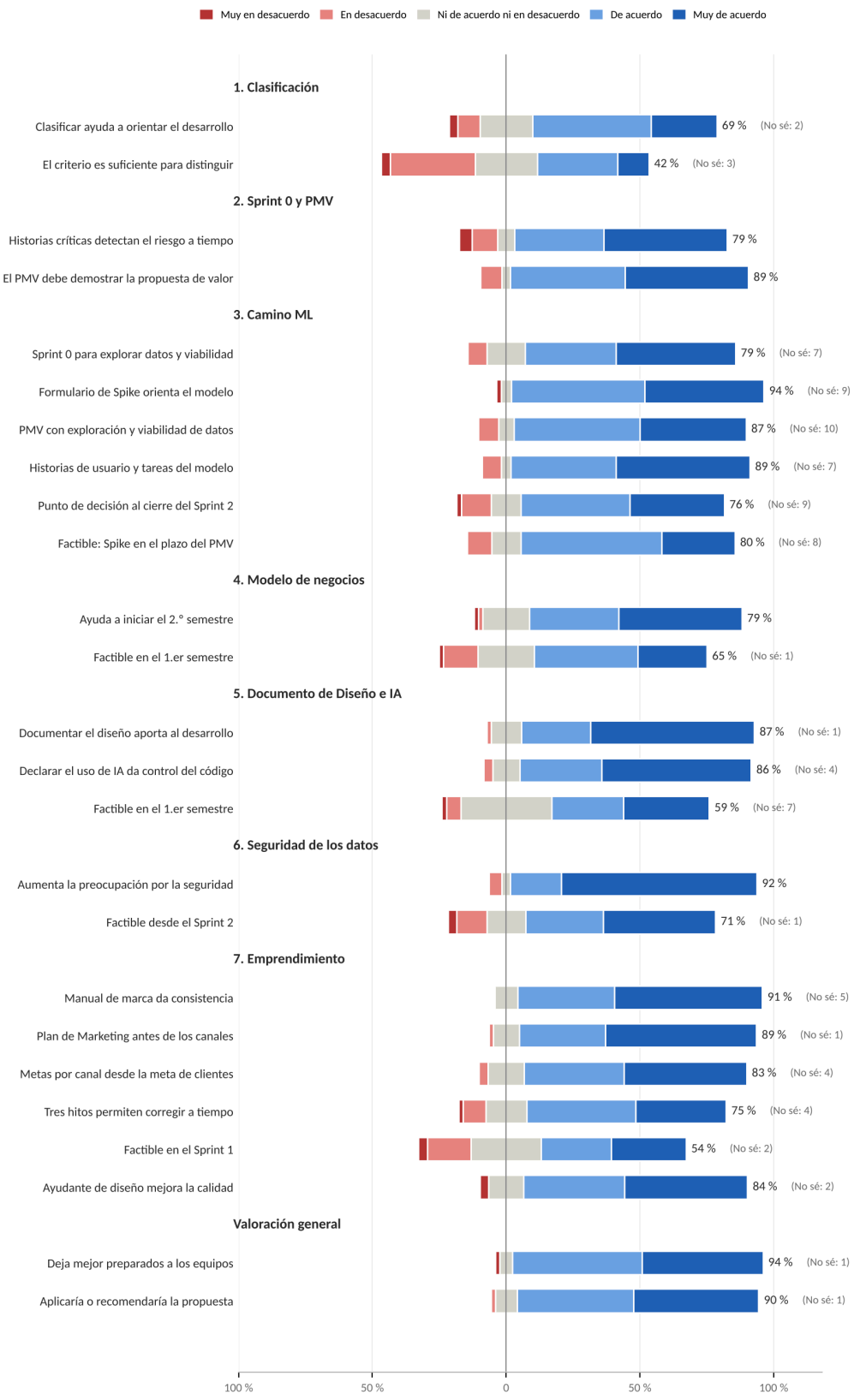


Figura 16: Respuestas a las afirmaciones de cada propuesta. El porcentaje indica las respuestas de acuerdo y muy de acuerdo sin contar «No sé». Fuente: Elaboración propia.

Afirmación	Total	Estudiantes	Ayudantes	Profesores
Clasificación por naturaleza del proyecto				
Clasificar ayuda a orientar el desarrollo	69 %	61 %	100 %	100 %
El criterio es suficiente para distinguir	42 %	40 %	50 %	50 %
Sprint 0 y presentación del PMV				
Historias críticas detectan el riesgo a tiempo	79 %	76 %	88 %	100 %
El PMV debe demostrar la propuesta de valor	89 %	86 %	100 %	100 %
Camino para proyectos de Machine Learning				
Sprint 0 para explorar datos y viabilidad	79 %	82 %	75 %	50 %
Formulario de Spike orienta el modelo	94 %	93 %	100 %	100 %
PMV con exploración y viabilidad de datos	87 %	88 %	100 %	50 %
Historias de usuario y tareas del modelo	89 %	87 %	100 %	100 %
Punto de decisión al cierre del Sprint 2	76 %	76 %	75 %	75 %
Factible realizar el Spike en el plazo del PMV	80 %	79 %	75 %	100 %
Modelo de negocios consolidado				
Ayuda a iniciar el segundo semestre	79 %	76 %	88 %	100 %
Factible en el primer semestre	65 %	64 %	62 %	75 %
Documento de Diseño y uso de IA				
Documentar el diseño aporta al desarrollo	87 %	88 %	75 %	100 %
Declarar el uso de IA da control del código	86 %	85 %	88 %	100 %
Factible en el primer semestre	59 %	57 %	67 %	75 %
Comprobación de seguridad de los datos				
Aumenta la preocupación por la seguridad	92 %	90 %	100 %	100 %

Afirmación	Total	Estudiantes	Ayudantes	Profesores
Factible desde el Sprint 2	71 %	70 %	62 %	100 %
Emprendimiento en el segundo semestre				
Manual de marca da consistencia	91 %	93 %	88 %	75 %
Plan de Marketing antes de los canales	89 %	88 %	88 %	100 %
Metas por canal desde la meta de clientes	83 %	85 %	88 %	50 %
Tres hitos permiten corregir a tiempo	75 %	73 %	86 %	75 %
Factible en el Sprint 1	54 %	52 %	71 %	50 %
Ayudante de diseño mejora la calidad	84 %	80 %	100 %	100 %
Valoración general				
Deja mejor preparados a los equipos	94 %	94 %	88 %	100 %
Aplicaría o recomendaría la propuesta	90 %	88 %	100 %	100 %

Tabla 21: Porcentaje de acuerdo por afirmación, para el total de participantes y por rol. Fuente: Elaboración propia.

Clasificación por naturaleza del proyecto. Es la única propuesta que queda no validada. El 69 % está de acuerdo con que clasificar el proyecto ayuda a orientar su desarrollo, pero solo el 42 % considera que el criterio es suficiente para distinguir ambos tipos de proyecto, y un 35 % está en desacuerdo. El rechazo al criterio no se concentra en un grupo, ya que aparece tanto en estudiantes con modelo propio como en los que integran una IA ya entrenada, y ayudantes y profesores se dividen en partes iguales. Las respuestas abiertas ayudan a entender el resultado. Dos estudiantes piden un rol dedicado al modelo en los proyectos de ML y otro interpreta que la propuesta «dejará a los proyectos tradicional de lado y se centran solo en ML e IA», lo que muestra que la clasificación se leyó como una separación entre mundos distintos y no como un camino que agrega actividades al desarrollo de software.

Sprint 0 y presentación del PMV. Queda validada. El 79 % está de acuerdo con que abordar las historias críticas en el Sprint 0 permite detectar a tiempo si el equipo puede construir lo que propuso, y el 89 % con que el PMV debe demostrar la propuesta de valor antes de continuar. Es, además, una de las propuestas con mayor acuerdo entre ayudantes y profesores.

Camino para proyectos de Machine Learning. Queda validada, con acuerdos entre 76 % y 94 % en sus seis afirmaciones. La opción «No sé» alcanza entre 7 y 10 respuestas, la mayoría de estudiantes que no trabajan con un modelo propio, y en ningún caso supera el 30 %. Entre

los estudiantes con modelo propio, que son quienes viven este camino, el acuerdo es aún mayor, con un 93 % para dedicar el Sprint 0 a explorar los datos y un 89 % para la factibilidad del Spike dentro del plazo actual. Entre los profesores aparecen dudas sobre dedicar el Sprint 0 a la exploración de datos y sobre presentar esa exploración en el PMV, ya que dos de los cuatro no expresan acuerdo, uno de ellos en desacuerdo y el otro en una posición neutral. En las respuestas abiertas, un profesor señala que lo más difícil será el seguimiento de los proyectos de ML y un ayudante advierte que conseguir datos reales antes del PMV suele retrasarse por impedimentos legales o por la poca participación del cliente.

Modelo de negocios consolidado. Queda validada con observaciones. El 79 % considera que contar con el modelo de negocios al cierre del primer semestre ayuda a iniciar el segundo con el negocio claro, pero la factibilidad de desarrollarlo junto al Documento de Diseño baja a 65 %. Las respuestas abiertas apuntan en dos direcciones que no se contradicen. Por un lado, varios estudiantes piden hablar antes de cómo cobrar por el producto, ya que «en todo el curso, hasta lo que llevamos actualmente que es un mes antes de feria, nunca nos hablan de como monetizar o cobrar por nuestro producto». Por otro, se advierte que es el PMV el que determina los costos reales y que el modelo debería trabajarse como «una noción ajustable» y no como una entrega definitiva.

Documento de Diseño y uso de IA. Queda validada con observaciones. El 87 % está de acuerdo con que documentar y justificar la arquitectura, el modelo de base de datos y los atributos de calidad aporta al desarrollo, y el 86 % con que declarar las herramientas de IA ayuda a mantener el control sobre el código. La factibilidad de completar las cuatro secciones nuevas en el primer semestre baja a 59 %. Un profesor advierte que los documentos de diseño del primer semestre pueden quedar obsoletos con los ajustes del segundo, lo que refuerza la necesidad de la Actualización de documentación en cada sprint, y un estudiante teme que, al aumentar las secciones, el equipo termine completándolas con IA. Al mismo tiempo, otra respuesta respalda directamente la propuesta, al señalar que «hay muchos grupos que cuentan solo con 1 desarrollador agentico y ningún integrante tiene noción alguna del proyecto en el código».

Comprobación de seguridad de los datos. Queda validada. El 92 % está de acuerdo con que incorporar esta comprobación haría que los equipos se preocupen más de la seguridad de sus proyectos y el 71 % la considera factible desde el Sprint 2. En las respuestas abiertas, un estudiante advierte la falta de conocimientos de programación segura con que llegan los equipos a la Feria, y otro plantea que en los proyectos con modelo propio el cuidado de los datos debería comenzar desde el Sprint 0.

Emprendimiento en el segundo semestre. Queda validada con observaciones. Las afirmaciones sobre la utilidad del proceso obtienen entre 75 % y 91 % de acuerdo, y el aporte del ayudante de diseño un 84 %. La factibilidad de desarrollar la marca, el Plan de Marketing y los canales durante el Sprint 1 es, en cambio, la afirmación con menor acuerdo entre las propuestas después del criterio de clasificación, con un 54 %, y un 43 % entre los estudiantes con modelo propio. Las respuestas abiertas lo explican. El Sprint 1 concentra «varias entregas

en simultaneo», y un estudiante propone ubicar el marketing «entre Sprint 1 y Sprint 2, ya que normalmente el Sprint 1 conlleva aplicar funcionalidades muy importantes para el producto». También se plantea que los canales deberían depender del tipo de cliente, ya que en un producto B2B el marketing digital no siempre es la vía por la que se captan clientes.

Valoración general. El 94 % considera que, en conjunto, la propuesta deja a los equipos mejor preparados para desarrollar su proyecto y continuarlo como emprendimiento, y el 90 % la aplicaría o recomendaría, completa o en parte, en un próximo semestre. Ambas afirmaciones superan el 85 % en los tres roles.

RESPUESTAS ABIERTAS

De los 63 participantes, 31 respondieron al menos una de las dos preguntas abiertas. El tema más repetido es la carga de trabajo, presente en 12 de esas respuestas. Los participantes no rechazan las actividades propuestas, sino que advierten que se suman a una asignatura que ya es exigente. Un estudiante señala que «muchas entregas (sobre todo en el primer semestre del ramo) estaban muy demás o eran prácticamente lo mismo que la anterior con un pequeño cambio o agregado», y un profesor resume la misma preocupación al indicar que sería «muy cuidadoso con el equilibrio de tareas, si se agregan algunas necesariamente deben reducirse otras». Otro profesor y un estudiante relacionan esta carga con el aumento de créditos de ambas asignaturas, y consideran que la propuesta calza mejor con el nuevo plan de estudios.

Los demás temas se reparten entre el momento en que se trabaja el negocio, con seis respuestas que piden abordar antes el tema del dinero o que advierten la dificultad de definirlo temprano, las particularidades de los proyectos de ML, con siete respuestas, y el marketing, con seis respuestas sobre su carga y su ubicación dentro del semestre. Algunas respuestas abordan temas que quedan fuera del alcance de este trabajo, como la dificultad de encontrar un cliente, el rol del cliente como validador del producto final, la subjetividad de algunas retroalimentaciones, el uso de Discord como canal oficial y la coordinación entre profesores. Estos temas se recogen como trabajo futuro.

4.4. SÍNTESIS DE LA VALIDACIÓN

La Tabla 22 reúne el resultado de cada propuesta en ambas instancias de validación y el ajuste o la recomendación que genera.

Propuesta	**Revisión con el 3IE**	**Encuesta**	**Ajuste o recomendación**
Clasificación por naturaleza del proyecto	No se revisó.	No validada (criterio 42 %).	Se ajusta la propuesta. Los caminos pasan a llamarse proyectos sin y con modelo propio, y se deja explícito que el segundo agrega las actividades de CRISP-DM sin reemplazar el desarrollo de software.
Sprint 0 y presentación del PMV	Valoró la estructura del primer semestre.	Validada.	Se mantiene.
Camino para proyectos de Machine Learning	No se revisó.	Validada.	Se mantiene.
Modelo de negocios consolidado	Se renombró y se adelantó al primer semestre.	Validada con observaciones (factibilidad 65 %).	Se mantiene. Se recomienda trabajarlo como una primera versión que el equipo actualiza durante el segundo semestre, a medida que conoce los costos reales de su producto.
Documento de Diseño y uso de IA	No se revisó.	Validada con observaciones (factibilidad 59 %).	Se mantiene. Se recomienda acotar la extensión de las cuatro secciones nuevas en su primera versión, considerando que el equipo las completa en la Actualización de documentación de cada sprint.
Comprobación de seguridad de los datos	No se revisó.	Validada.	Se mantiene.
Emprendimiento en el segundo semestre	Se eliminó el mapa del recorrido del cliente.	Validada con observaciones (factibilidad 54 %).	Se mantiene. Se recomienda que la marca y el Plan de Marketing se desarrollen en el Sprint 1 y que la puesta en marcha de los canales se extienda hacia el Sprint 2.

Tabla 22: Síntesis de la validación de cada propuesta. Fuente: Elaboración propia.

En conjunto, seis de las siete propuestas superan el 70 % de acuerdo en todas sus afirmaciones de utilidad, y en las tres que quedan con observaciones el punto débil es siempre la factibilidad, no la utilidad. Esto indica que los participantes reconocen el valor de los ajustes, pero que su preocupación está en cómo calzan dentro de los tiempos del curso. Por esta razón, las observaciones de factibilidad no se traducen en cambios a la propuesta, sino en recomendaciones de implementación dirigidas a los profesores, quienes calibran cada semestre la exigencia y los plazos de las entregas. Esta calibración debe considerar que la propuesta no elimina entregables del curso, como muestra la Tabla 23 en los anexos, y que el aumento de créditos previsto para GPI y TDPI en el nuevo plan de estudios es el contexto en el que esa carga adicional puede absorberse, en especial en la parte de emprendimiento.

La única propuesta que se modifica es la clasificación por naturaleza del proyecto, ya que su resultado no corresponde a un problema de factibilidad sino de claridad. La encuesta se respondió con la versión anterior de los nombres de ambos caminos, que distinguía entre el camino de software tradicional y el camino de ML, y las respuestas abiertas muestran que esa distinción se leyó como una separación entre tipos de proyecto opuestos. El ajuste, que coincide además con una observación de la profesora correferente Luz Chourio durante la revisión de esta memoria, mantiene el criterio, pero cambia cómo se presenta.

4.4.1. LIMITACIONES

La validación tiene dos limitaciones. Primero, mide la percepción de los actores sobre la propuesta y no sus resultados en un semestre real, ya que la implementación como piloto queda fuera del alcance de este trabajo. Segundo, la encuesta depende de la participación voluntaria, por lo que el número de respuestas por rol puede ser acotado y sus resultados deben leerse como una tendencia y no como una medición representativa. En esta aplicación respondieron 51 estudiantes, 8 ayudantes y 4 de los 6 profesores de las asignaturas, por lo que las diferencias entre roles se presentan como indicios y no como diferencias concluyentes.

CAPÍTULO 5
CONCLUSIONES

Este trabajo tuvo como objetivo proponer una adecuación del flujo de procesos ágil de GPI y TDPI que responda a la realidad actual de sus proyectos y de las herramientas que se utilizan para desarrollarlos. El trabajo partió de la constatación de que el marco definido por Calvo [1] en 2019 sigue siendo una buena base, pero fue pensado para un tipo de proyecto y una forma de desarrollar que ya no son las predominantes en la Feria de Software. Por eso, la propuesta no reemplaza ese marco, sino que lo adecua. Conserva sus ceremonias, sus roles y su calendario, y concentra los cambios en el contenido de los artefactos que los equipos ya entregan y en algunos procesos que hoy no existen.

5.1. CUMPLIMIENTO DE LOS OBJETIVOS

El primer objetivo específico se cumplió mediante el análisis del documento de 2019 y la experiencia de la autora como estudiante y como ayudante de ambas asignaturas. Este diagnóstico permitió identificar cinco puntos de desajuste en el flujo vigente, que son la realización del prototipo, la exigencia de un producto funcional al cierre de cada sprint, que no deja espacio para el avance no lineal de los proyectos con modelo propio, la falta de exigencias sobre la documentación técnica, la ausencia de un proceso definido de emprendimiento y la falta de un punto de control sobre el manejo de datos personales. Un hallazgo común a todos ellos es que no se trata de casos aislados, sino de problemas que se mantienen porque ningún hito del flujo actual exige resolverlos.

El segundo objetivo se abordó con el levantamiento de los proyectos presentados en la Feria de Software entre 2018 y 2025 y con la revisión de estudios recientes sobre el uso de IA en el desarrollo de software. El levantamiento mostró que los proyectos con algún componente de inteligencia artificial pasaron de ser la excepción en 2018, con 3 de 16 proyectos, a ser la mayoría entre 2023 y 2025, con 64 de 94. Los estudios revisados muestran, por su parte, que el uso de herramientas de IA ya es mayoritario entre desarrolladores y estudiantes, y que cambia la forma de trabajar desde escribir código hacia supervisar lo que la herramienta genera, con el riesgo de que la documentación y las decisiones de diseño queden en segundo plano.

El tercer objetivo se cumplió con la revisión de los marcos ágiles vigentes en la industria. Esta revisión mostró que el marco de Calvo [1] coincide en lo esencial con la práctica actual, ya que la mayoría de las organizaciones trabaja hoy con modelos híbridos o adaptados de Scrum, pero que no contempla el uso de herramientas de IA dentro del flujo de desarrollo. Junto con la revisión de los modelos de proceso de datos del marco conceptual, esto llevó a mantener Scrum como base y a incorporar CRISP-DM solo como referencia para el trabajo de los proyectos con modelo propio.

El cuarto objetivo se cumplió con la propuesta documentada en el capítulo 3, en la que cada proceso se describe con su objetivo, sus entradas y salidas, sus reglas de negocio y su diagrama. Los ajustes principales son la clasificación de los proyectos según su naturaleza, con un Sprint 0 mediante un Spike para los proyectos con modelo propio y la integración de las fases de CRISP-DM en sus sprints, un PMV que debe demostrar la propuesta de valor del proyecto, el reemplazo del Plan de Proyecto por un Documento de Diseño que se mantiene actualizado durante todo el año, la comprobación de seguridad de los datos desde el Sprint 2 y una toma de decisiones al cierre de ese sprint que permite ajustar el alcance del objetivo de Machine Learning. A esto se suma un proceso de emprendimiento que parte con el modelo de negocios consolidado al cierre del primer semestre y continúa en el segundo con el diseño de la marca, el Plan de Marketing y tres Reportes de Marketing.

El quinto objetivo se cumplió en dos instancias. La revisión con el 3IE valoró positivamente la estructura del primer semestre y llevó a ajustar la propuesta antes de aplicar la encuesta. A partir de ella se adelantó el modelo de negocios al primer semestre, se cambió su nombre para que no sugiriera un documento más extenso de lo que se pide y se eliminó el mapa del recorrido del cliente para reducir la carga del segundo semestre. La encuesta reunió 63 respuestas de estudiantes, ayudantes y cuatro de los seis profesores de las asignaturas. Seis de las siete propuestas quedaron validadas o validadas con observaciones, y el 90 % de los participantes aplicaría o recomendaría la propuesta en un próximo semestre. Las propuestas de Sprint 0 y PMV, del camino para proyectos con modelo propio y de la comprobación de seguridad de los datos quedaron validadas. Las del modelo de negocios consolidado, el Documento de Diseño y el emprendimiento del segundo semestre quedaron validadas con observaciones, siempre por su factibilidad y no por su utilidad, por lo que esas observaciones se plantean como recomendaciones de implementación para los profesores. La clasificación por naturaleza del proyecto fue la única no validada, porque su criterio no se entendió como un camino que amplía el desarrollo de software, y se ajustó la forma de presentarla. En cuanto al diagnóstico, los actores reconocieron los desajustes concretos, como el efecto de la IA, la desconexión del marketing y el trabajo de los proyectos de ML, mientras que la percepción general de que el flujo no se ajusta se concentró en los estudiantes.

5.2. APORTES DEL TRABAJO

El principal aporte de este trabajo es un documento de referencia actualizado del flujo de procesos de la Feria de Software, que profesores y ayudantes pueden usar para organizar ambas asignaturas y que deja explícito qué se espera de los equipos en cada etapa. Esto responde directamente a una de las causas identificadas en el diagnóstico, ya que muchos de los problemas observados se mantenían porque no estaban definidos en ningún hito.

Un segundo aporte es la forma en que se abordó la diversidad de proyectos. En lugar de exigir lo mismo a todos los equipos o de crear un flujo distinto para cada tipo de proyecto, la propuesta mantiene un único flujo y solo diferencia lo que cada naturaleza de proyecto

necesita, como el contenido del Sprint 0, del PMV y del Documento de Diseño. Este criterio puede servir para incorporar en el futuro otros tipos de proyecto sin tener que rediseñar el curso completo.

Por último, la propuesta acerca la Feria de Software a lo que pide el nuevo plan de estudios, que incorpora la innovación y el emprendimiento como parte de lo que el estudiante debe lograr. El proceso de emprendimiento deja a cada equipo, al terminar el año, con un modelo de negocios, una marca y un registro de cómo funcionaron sus canales, que le sirve como base si decide continuar con su proyecto.

5.3. LIMITACIONES

La principal limitación es que la propuesta no se implementó durante el desarrollo de este trabajo, por lo que su validación mide la percepción de los actores y no sus resultados en un semestre real. A esto se suma que la encuesta dependió de la participación voluntaria, por lo que sus resultados deben leerse como una tendencia.

El diagnóstico también tiene límites. La experiencia de la autora corresponde a una sola mirada dentro del curso, y la clasificación de los proyectos de la Feria se basa en lo que cada equipo declara sobre su proyecto y no en una revisión de su implementación. Además, la propuesta clasifica los proyectos en solo dos naturalezas, por lo que los proyectos que integran un modelo ya entrenado, como un modelo de lenguaje mediante un servicio externo, siguen el camino sin modelo propio aunque puedan tener necesidades propias.

Finalmente, la propuesta define el proceso, es decir, qué se hace, quién lo hace y qué se entrega en cada etapa, pero no cómo se evalúa cada entregable. Las pautas y rúbricas de evaluación quedan fuera de su alcance.

5.4. TRABAJO FUTURO

El paso más directo es implementar la propuesta como piloto en una versión de GPI y TDPI, para medir si los ajustes resuelven los desajustes identificados. En particular, convendría observar si el Documento de Diseño se mantiene realmente actualizado, si el reparto entre historias de usuario y tareas de modelado funciona para equipos de distinto tamaño y si los Reportes de Marketing permiten a los equipos tomar decisiones sobre sus canales.

También queda pendiente definir las pautas de evaluación de los nuevos entregables, definir los controles de cumplimiento frente a la Ley 21.719 una vez que entre en vigencia y automatizar la comprobación de seguridad de los datos mediante scripts que apoyen la revisión de los equipos. Un piloto permitiría, además, evaluar las recomendaciones de implementación que dejó la encuesta, en particular la distribución de las actividades de marketing entre los Sprints 1 y 2 y el trabajo del modelo de negocios como una versión que se actualiza durante el año. Algunos temas que surgieron en las respuestas abiertas quedan

también como trabajo futuro, como el rol del cliente como validador del producto final y la dificultad de los equipos para encontrar un cliente y un problema acordes a la exigencia del curso. Por último, se recomienda repetir el levantamiento de proyectos de la Feria cada cierto tiempo, para detectar a tiempo nuevos cambios en el tipo de proyecto que llega al curso. Si los proyectos que integran modelos ya entrenados siguen creciendo, podría ser necesario evaluar si requieren un tratamiento propio dentro del flujo.

5.5. REFLEXIÓN FINAL

Haber pasado por la Feria de Software primero como estudiante y luego como ayudante permitió ver el mismo flujo desde dos lugares distintos, y entender que muchos de sus problemas no se deben a los equipos ni a quienes dirigen el curso, sino a que el marco no había vuelto a revisarse desde que se definió. El trabajo de Calvo [1] mostró en su momento que adaptar Scrum a un contexto universitario era posible, y este trabajo buscó continuar esa misma lógica. La Feria de Software va a seguir cambiando junto con los proyectos y las herramientas, por lo que el principal aprendizaje de este trabajo es que su flujo de procesos necesita revisarse de forma periódica para seguir respondiendo a la realidad de sus estudiantes.

ANEXOS

Plantillas del 3IE utilizadas como Lean Canvas

A continuación se presentan las plantillas del Instituto Internacional para la Innovación Empresarial (3IE) que, para efectos de este trabajo, conforman el Lean Canvas de cada proyecto.



Figura 17: Template del Problema. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM.



LEAN CANVAS · TALLER PRESENCIAL

TEMPLATE DE SEGMENTO DE CLIENTES

EQUIPO

Integrantes: _____

Primero definan si venden a personas u organizaciones; luego delimiten un segmento específico, localizable y cuantificable.

1 DEFINAN EL MODELO Y LOS ROLES

En B2B, usuario, comprador y decisor pueden ser personas distintas.

1 MODELO DE NEGOCIO
Marquen: ☐ B2C ☐ B2B ☐ HÍBRIDO. ¿Quién usa y quién paga?

2 USUARIO / BENEFICIARIO
¿Quién utiliza la solución o recibe directamente el beneficio? Describanlo con precisión.

3 COMPRADOR Y DECISOR
¿Quién paga? ¿Quién decide, influye o autoriza la compra?

2 DELIMITEN EL SEGMENTO PRIORITARIO

CRITERIOS DE SEGMENTACIÓN
B2C: ubicación, edad y conducta. B2B: industria, tamaño, cobertura y digitalización.

SEGMENTO PRIORITARIO
¿Qué grupo comparte el problema, la urgencia y una forma similar de comprar?

DÓNDE ENCONTRARLOS
Ciudades, empresas, plataformas, asociaciones, eventos, bases de datos o comunidades.

3 DIMENSIONEN EL MERCADO
Usen una fuente o expliciten el supuesto.

TAM · MERCADO TOTAL
Todos los clientes que podrían necesitar la solución.
Cantidad: _____
Fuente: _____

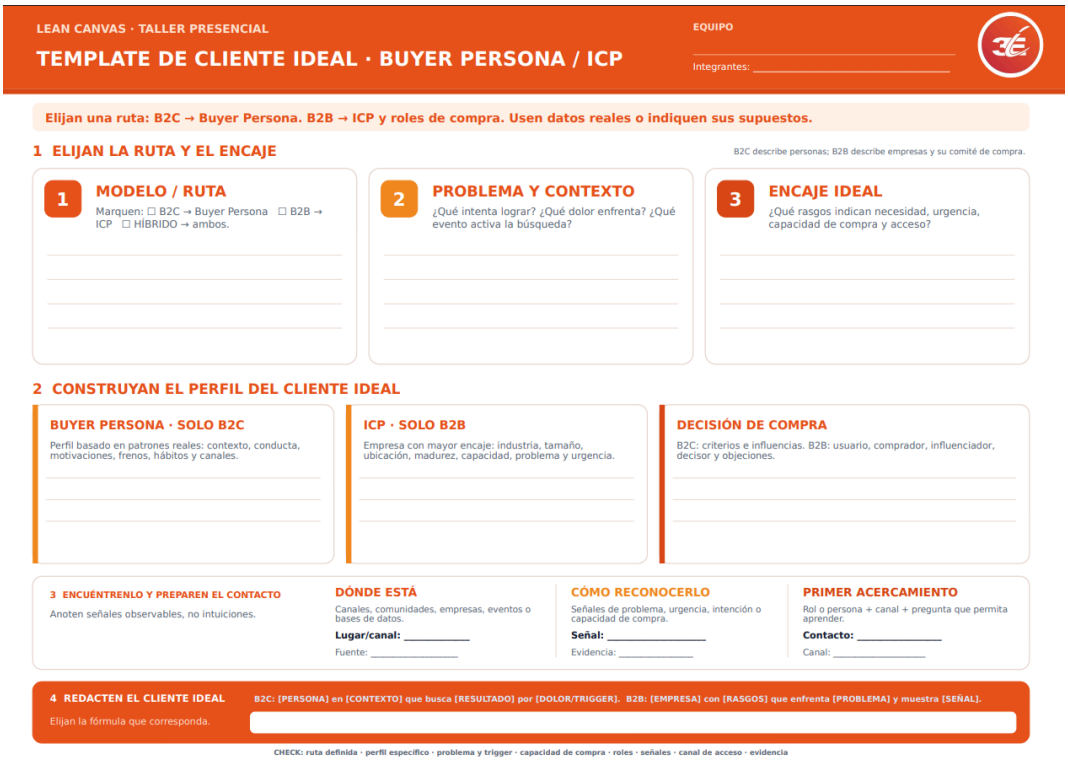
SAM · MERCADO ATENDIBLE
Parte del TAM que podemos servir por foco y capacidad.
Cantidad: _____
Criterio: _____

SOM · MERCADO ALCANZABLE
Clientes que podemos captar de manera realista al inicio.
Cantidad: _____
Horizonte: _____ meses

4 REDACCIÓN DEL SEGMENTO
Una frase específica y comprobable.
Nuestro segmento prioritario es [CLIENTE] en [LUGAR/CANAL], que presenta [PROBLEMA] y puede ser alcanzado mediante [CANAL].

CHECK: modelo definido · usuario y comprador diferenciados · segmento específico · localizable · cuantificable · SOM realista

Figura 18: Template de Segmento de Clientes. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM.



LEAN CANVAS · TALLER PRESENCIAL

TEMPLATE DE CLIENTE IDEAL · BUYER PERSONA / ICP

EQUIPO

Integrantes: _____

Elijan una ruta: B2C → Buyer Persona. B2B → ICP y roles de compra. Usen datos reales o indiquen sus supuestos.

1 ELIJAN LA RUTA Y EL ENCAJE

B2C describe personas; B2B describe empresas y su comité de compra.

1 MODELO / RUTA
Marquen: ☐ B2C → Buyer Persona ☐ B2B → ICP ☐ HÍBRIDO → ambos.

2 PROBLEMA Y CONTEXTO
¿Qué intenta lograr? ¿Qué dolor enfrenta? ¿Qué evento activa la búsqueda?

3 ENCAJE IDEAL
¿Qué rasgos indican necesidad, urgencia, capacidad de compra y acceso?

2 CONSTRUYAN EL PERFIL DEL CLIENTE IDEAL

BUYER PERSONA · SOLO B2C
Perfil basado en patrones reales: contexto, conducta, motivaciones, frenos, hábitos y canales.

ICP · SOLO B2B
Empresa con mayor encaje: industria, tamaño, ubicación, madurez, capacidad, problema y urgencia.

DECISIÓN DE COMPRA
B2C: criterios e influencias. B2B: usuario, comprador, influenciador, decisor y objeciones.

3 ENCUÉNTRENLO Y PREPAREN EL CONTACTO
Anoten señales observables, no intuiciones.

DÓNDE ESTÁ
Canales, comunidades, empresas, eventos o bases de datos.
Lugar/canal: _____
Fuente: _____

CÓMO RECONOCERLO
Señales de problema, urgencia, intención o capacidad de compra.
Señal: _____
Evidencia: _____

PRIMER ACERCAMIENTO
Rol o persona + canal + pregunta que permita aprender.
Contacto: _____
Canal: _____

4 REDACTEN EL CLIENTE IDEAL
Elijan la fórmula que corresponda.
B2C: [PERSONA] en [CONTEXTO] que busca [RESULTADO] por [DOLOR/TRIGGER]. B2B: [EMPRESA] con [RASGOS] que enfrenta [PROBLEMA] y muestra [SEÑAL].

CHECK: ruta definida · perfil específico · problema y trigger · capacidad de compra · roles · señales · canal de acceso · evidencia

Figura 19: Template de Cliente Ideal. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM.



Figura 20: Template de la Solución. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM.



Figura 21: Template de Propuesta de Valor. Fuente: Instituto Internacional para la Innovación Empresarial (3IE), UTFSM.

Proceso de emprendimiento del segundo semestre

A continuación se presenta en formato horizontal el diagrama del proceso de emprendimiento del segundo semestre (ver Figura 14), para facilitar su lectura.

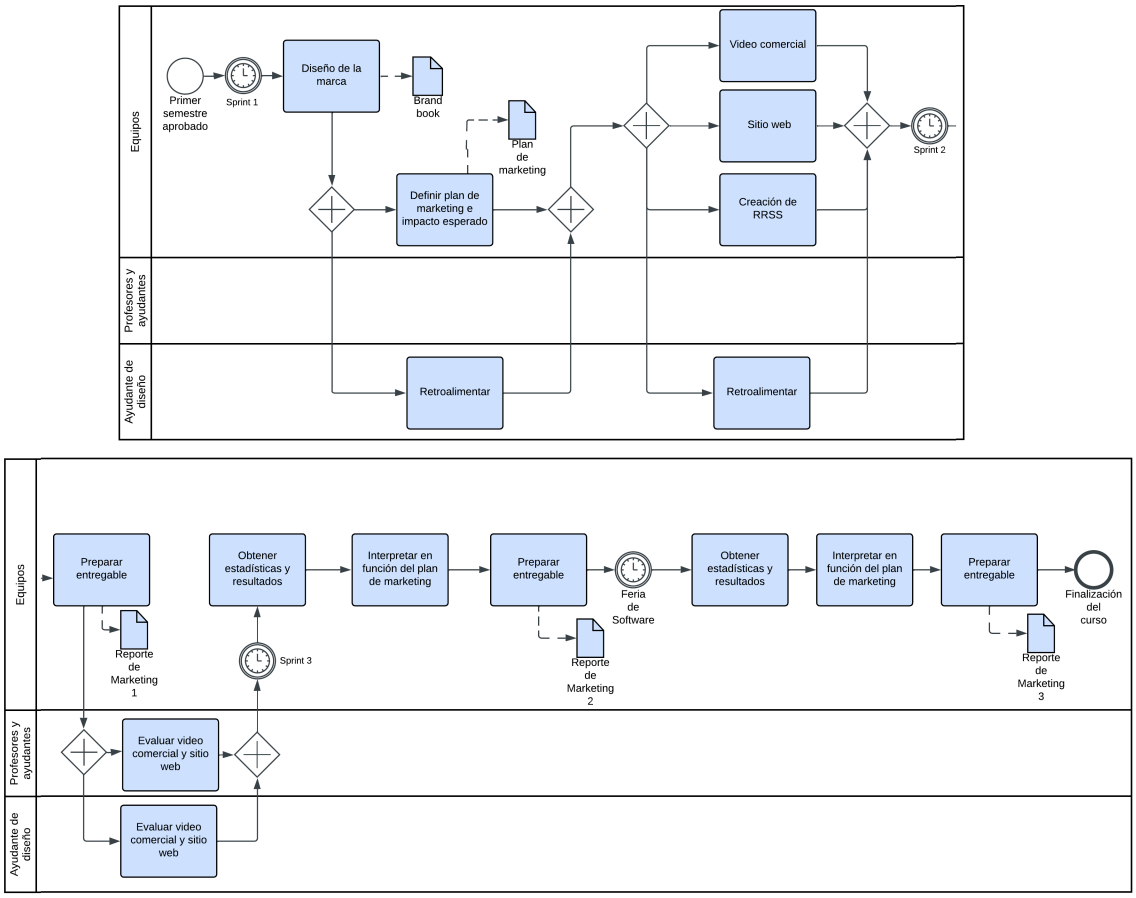


Figura 22: Proceso de emprendimiento del segundo semestre, versión dividida. Fuente: Elaboración propia.

Sprints combinados con el proceso de emprendimiento

A continuación se presentan los diagramas de los Sprints 1, 2 y 3 combinados con las actividades del proceso de emprendimiento (ver *Emprendimiento*), para mostrar en qué momento de cada sprint ocurren. Las actividades de emprendimiento se identifican en azul y las de desarrollo en amarillo.

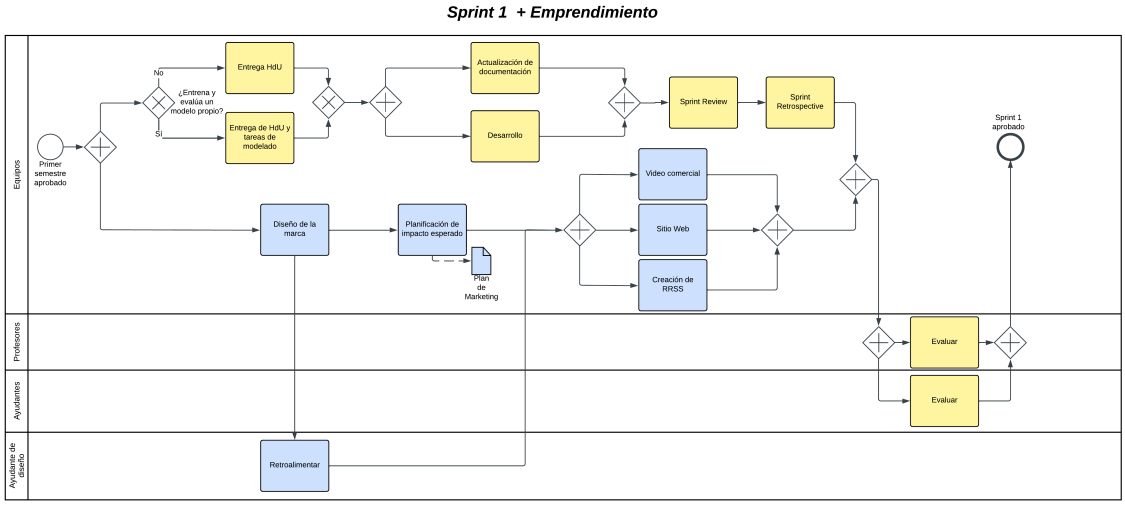


Figura 23: Sprint 1 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia.

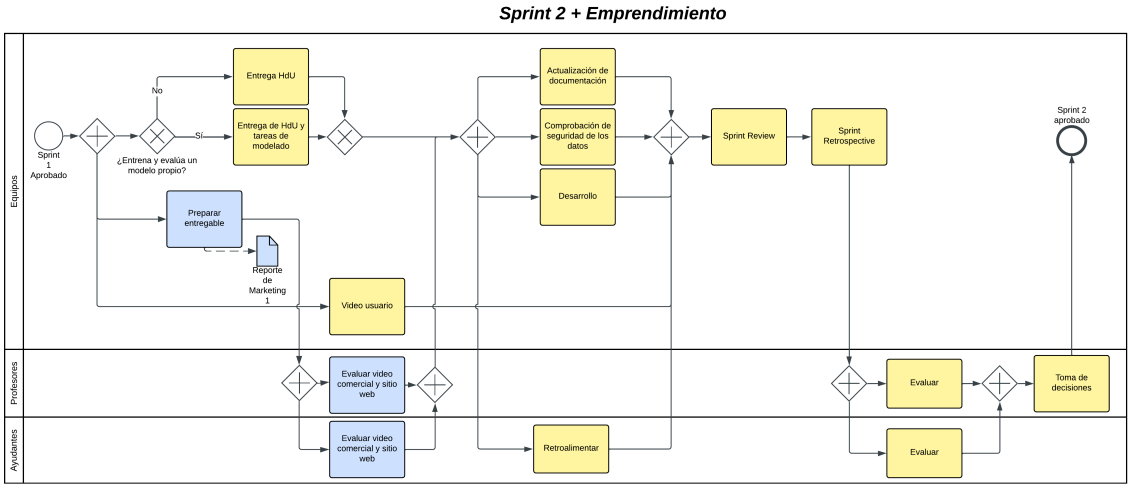


Figura 24: Sprint 2 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia.

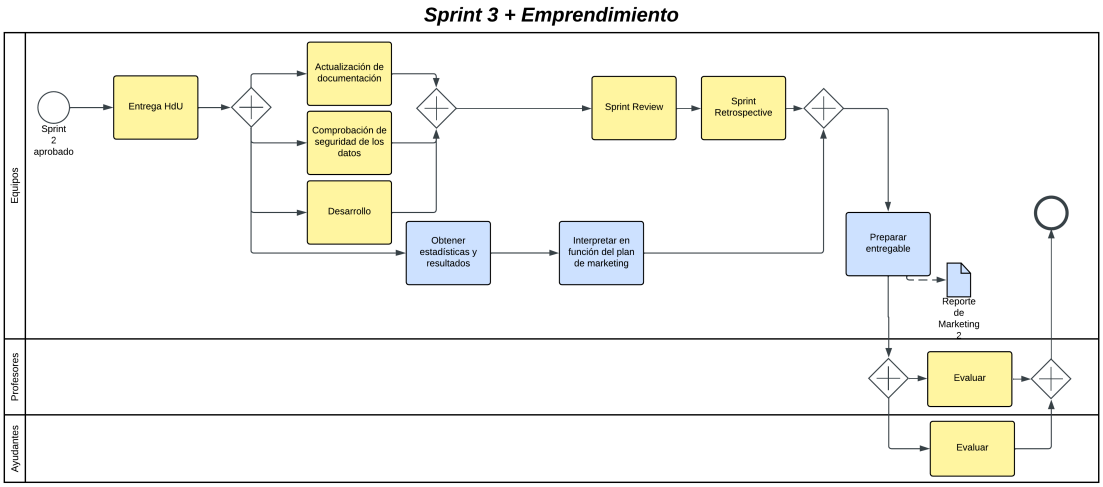


Figura 25: Sprint 3 combinado con las actividades del proceso de emprendimiento. Fuente: Elaboración propia.

Cambios de la propuesta respecto del curso actual

A continuación se resume, para cada entregable y actividad del flujo, qué cambia respecto del marco vigente o de cómo se realiza hoy el curso. La tabla distingue los elementos que se mantienen, los que se modifican en su contenido o criterio, los que reemplazan a un entregable existente, los que cambian de momento dentro del año y los que se agregan. Cuando un elemento queda a cargo de profesores o ayudantes se indica en la columna de cambio, ya que en esos casos no suma trabajo para el equipo.

Elemento	**Marco vigente o práctica actual**	**En la propuesta**	**Cambio**
Primer semestre			
Etapas de innovación	Cuatro etapas definidas por Cayo [2].	Se formaliza su entrega a partir de las plantillas del 3IE.	Se formaliza
Documento de propuesta y Licitación	Definidos en el marco vigente.	Sin cambios.	Se mantiene
Clasificación por naturaleza del proyecto	No existe.	Profesores y ayudantes clasifican el proyecto después de la Licitación.	Se agrega, a cargo de profesores y ayudantes
Formulario del Sprint 0 en proyectos sin modelo propio	Formulario de Historias de Usuario con historias de alto valor y menor dificultad.	Mismo formulario, con las historias críticas del proyecto.	Se modifica

Elemento	**Marco vigente o práctica actual**	**En la propuesta**	**Cambio**
Formulario del Sprint 0 en proyectos con modelo propio	Formulario de Historias de Usuario.	Formulario de Spike.	Se reemplaza
Prototipo inicial	Muestra historias de alto valor y menor dificultad.	PMV que debe demostrar la propuesta de valor.	Se modifica
Modelo de negocios	Business Model Canvas y modelo de negocio preliminar después del Sprint 2.	Modelo de negocios consolidado al cierre del primer semestre.	Se reemplaza y se adelanta
Plan de Proyecto	Plan de Proyecto.	Documento de Diseño, que conserva sus secciones y agrega arquitectura, modelo de base de datos, atributos de calidad y uso de IA.	Se reemplaza y se amplía
Segundo semestre			
Formulario de entrega del sprint	Historias de usuario comprometidas con sus puntos de historia.	Mismo formulario, con una sección de cambios respecto de lo planificado y, en proyectos con modelo propio, tareas de modelado y reparto de capacidad.	Se modifica
Actualización de documentación	No existe.	Revisión del Documento de Diseño en cada sprint.	Se agrega
Comprobación de seguridad de los datos	No existe.	Revisión del tratamiento de los datos sensibles desde el Sprint 2.	Se agrega
Video usuario	Se realiza en el Sprint 2.	Sin cambios.	Se mantiene
Evento Feria de Software	Exposición del producto y pitch en el Demoday.	Sin cambios.	Se mantiene
Documento de Legado	Documento grupal.	Conserva su estructura y agrega las lecciones aprendidas sobre los elementos de la propuesta.	Se modifica
Emprendimiento			

Elemento	**Marco vigente o práctica actual**	**En la propuesta**	**Cambio**
Manual de marca	No existe como entregable.	Manual de marca al inicio del Sprint 1.	Se agrega
Ayudante de diseño	Se ha usado en las últimas versiones del curso sin estar formalizado.	Retroalimenta la marca, el video comercial y el sitio web.	Se formaliza
Plan de Marketing	Los canales se construyen sin metas definidas.	Plan con KPI y metas por canal, calculadas a partir de la meta de clientes.	Se agrega
Reportes de marketing	Tres reportes de marketing repartidos de otra forma y sin un Plan de Marketing que los conecte.	Reporte de Marketing 1 en el Sprint 2, Reporte de Marketing 2 en el Sprint 3 y Reporte de Marketing 3 después de la Feria.	Se modifica

Tabla 23: Cambios de la propuesta respecto del marco vigente y de la práctica actual del curso. Fuente: Elaboración propia.

La propuesta no elimina entregables del curso. Los elementos que reemplazan a un entregable existente se concentran en el primer semestre, con el Documento de Diseño y el modelo de negocios consolidado. Lo que se agrega, en cambio, se concentra en el proceso de Emprendimiento y en las actividades paralelas de los sprints. Esta distribución es la que debe considerarse al calibrar la carga de cada instancia, en particular frente al aumento de créditos de GPI y TDPI previsto en el nuevo plan de estudios.

REFERENCIAS BIBLIOGRÁFICAS

[1] R. Á. Calvo Erices, «Propuesta basada en métodos ágiles para desarrollo de proyectos capstone de emprendimiento e innovación», Memoria de titulación para optar al título de Ingeniero Civil Informático, Santiago, Chile, 2019.

[2] B. A. Cayo Vidal, «Propuesta de marco de trabajo para la innovación tecnológica en proyectos de informática de Feria de Software de la UTFSM», Memoria para optar al título de Ingeniero Civil en Informática, Valparaíso, Chile, 2025.

[3] R. Wirth y J. Hipp, «CRISP-DM: Towards a Standard Process Model for Data Mining», en *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 2000, pp. 29-39.

[4] International Engineering Alliance, «Graduate Attributes and Professional Competencies», Informe técnico, 2021.

[5] R. H. Todd, S. P. Magleby, C. D. Sorensen, B. R. Swan, y D. K. Anthony, «A Survey of Capstone Engineering Courses in North America», *Journal of Engineering Education*, vol. 84, n.° 2, pp. 165-174, 1995, doi: 10.1002/j.2168-9830.1995.tb00163.x.

[6] G. C. Ng, «A Study of an Agile Methodology with Scrum Approach to the Filipino Company-Sponsored I.T. Capstone Program», *International Journal of Computing Sciences Research*, vol. 2, n.° 2, pp. 68-88, 2018, doi: 10.25147/ijcsr.2017.001.1.25.

[7] N. Damianova y S. Berrezueta-Guzman, «The Anatomy of a Successful Student Scrum Team: Motivation, Personalities, and Academic Adaptation». 2026.

[8] S. Martín, «Aplicación de las Metodologías Ágiles al proceso de enseñanza-aprendizaje universitario», *RIDU Revista d'Innovació Docent Universitaria*, vol. 12, pp. 62-73, 2020.

[9] Naciones Unidas, «La Agenda 2030 y los Objetivos de Desarrollo Sostenible». [En línea]. Disponible en: https://www.un.org/sustainabledevelopment/es/sustainable--development-goals/

[10] D. A. Kolb, *Experiential Learning: Experience as the Source of Learning and Development*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[11] Feria de Software USM, «Ediciones y proyectos de la Feria de Software». [En línea]. Disponible en: https://www.feriadesoftware.cl/ediciones

[12] J. Mateluna Cuadra, «Estudiantes USM presentan soluciones tecnológicas en la 33ª Feria de Software». [En línea]. Disponible en: https://www.cooperativaciencia.cl/tecnologia/2025/11/12/estudiantes-usm-presentan-soluciones-tecnologicas-en-la-33a-feria-de-software/

[13] K. Schwaber y J. Sutherland, «La Guía Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego». [En línea]. Disponible en: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf

[14] Departamento de Informática, Universidad Técnica Federico Santa María, «Programa de Asignatura: Gestión de Proyectos de Informática (INF-360)», Valparaíso, Chile, Programa de asignatura, 2026.

[15] Departamento de Informática, Universidad Técnica Federico Santa María, «Programa de Asignatura: Taller de Desarrollo de Proyectos de Informática», Valparaíso, Chile, Programa de asignatura, 2026.

[16] Stack Overflow, «2025 Stack Overflow Developer Survey». [En línea]. Disponible en: https://survey.stackoverflow.co/2025/ai

[17] JetBrains, «The State of Developer Ecosystem 2025». [En línea]. Disponible en: https://www.jetbrains.com/lp/devecosystem-2025/ai/

[18] S. Marken, «AI Is Routine for College Students, Despite Campus Limits». [En línea]. Disponible en: https://news.gallup.com/poll/704090/routine-college-students--despite-campus-limits.aspx

[19] Project Management Institute, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 7.ª ed. Newtown Square, PA: Project Management Institute, 2021.

[20] W. W. Royce, «Managing the Development of Large Software Systems», en *Proceedings of IEEE WESCON*, 1970, pp. 1-9.

[21] K. Beck *et al.*, «Manifesto for Agile Software Development». [En línea]. Disponible en: https://agilemanifesto.org/

[22] L. M. Chourio y A. Pereira Vale, «Historias de Usuarios: Material de Cátedra». Valparaíso, Chile, 2026.

[23] S. Jain, V. Jotwani, y S. Agrawal, «Comparative Analysis of Agile Frameworks: Evaluating their Impact on Software Development Lifecycle Efficiency», *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, n.° 23s, pp. 3126-3137, 2024, [En línea]. Disponible en: https://ijisae.org/index.php/IJISAE/article/view/7627

[24] Digital.ai, «18th State of Agile Report». [En línea]. Disponible en: https://digital.ai/resource-center/analyst-reports/state-of-agile-report/

[25] J. W. Thomas, «A Review of Research on Project-Based Learning», San Rafael, CA, Informe técnico, 2000.

[26] S. Lavado-Anguera, P.-J. Velasco-Quintana, y M.-J. Terrón-López, «Project-Based Learning (PBL) as an Experiential Pedagogical Methodology in Engineering Education: A Review of the Literature», *Education Sciences*, vol. 14, n.° 6, p. 617, 2024, doi: 10.3390/educsci14060617.

[27] S. Tenhunen, T. Männistö, M. Luukkainen, y P. Ihantola, «A Systematic Literature Review of Capstone Courses in Software Engineering», *Information and Software Technology*, vol. 159, p. 107191, 2023, doi: 10.1016/j.infsof.2023.107191.

[28] V. Mahnic, «A Capstone Course on Agile Software Development Using Scrum», *IEEE Transactions on Education*, vol. 55, n.° 1, pp. 99-106, 2012, doi: 10.1109/TE.2011.2142311.

[29] Z. Masood, R. Hoda, y K. Blincoe, «Adapting Agile Practices in University Contexts», *Journal of Systems and Software*, vol. 144, pp. 501-510, 2018, doi: 10.1016/j.jss.2018.07.011.

[30] OECD y Eurostat, *Oslo Manual 2018: Guidelines for Collecting, Reporting and Using Data on Innovation*, 4.ª ed. Paris / Luxembourg: OECD Publishing / Eurostat, 2018. doi: 10.1787/9789264304604-en.

[31] S. Shane y S. Venkataraman, «The Promise of Entrepreneurship as a Field of Research», *Academy of Management Review*, vol. 25, n.° 1, pp. 217-226, 2000, doi: 10.2307/259271.

[32] S. Blank y B. Dorf, *The Startup Owner's Manual: The Step-by-Step Guide for Building a Great Company*. Pescadero, CA: K&S Ranch, 2012.

[33] Universidad Técnica Federico Santa María, «Modelo Educativo Institucional», Valparaíso, Chile, Documento institucional, 2016.

[34] C. R. Carlson y W. W. Wilmot, *Innovation: The Five Disciplines for Creating What Customers Want*. New York: Crown Business, 2006.

[35] A. Osterwalder y Y. Pigneur, *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. Hoboken, NJ: John Wiley & Sons, 2010.

[36] H. Sarooghi, S. Sunny, J. Hornsby, y S. Fernhaber, «Design Thinking and Entrepreneurship Education: Where Are We, and What Are the Possibilities?», *Journal of Small Business Management*, vol. 57, n.° sup1, pp. 78-93, 2019, doi: 10.1111/jsbm.12541.

[37] N. Duval-Couetil, A. Epstein, y A. Huang-Saad, «Catalyzing U.S. Innovation and Entrepreneurship: Approaching Assessment and Evaluation of the National Science Foundation's I-Corps Program», en *ASEE Annual Conference & Exposition*, 2022.

[38] Departamento de Informática, Universidad Técnica Federico Santa María, «Actividad E7: Business Model Canvas y Modelo de Negocio Preliminar. Rúbrica de evaluación». 2026.

[39] A. Maurya, *Running Lean: Iterate from Plan A to a Plan That Works*, 2.ª ed. Sebastopol, CA: O'Reilly Media, 2012.

[40] P. Godoy Barrera y M. Visconti Zamora, «Plan de Marketing: Material de Cátedra». 2026.

[41] R. Sapkota, K. I. Roumeliotis, y M. Karkee, «Vibe Coding vs. Agentic Coding: Fundamentals and Practical Implications of Agentic AI». 2025.

[42] A. Vella y K. Blincoe, «The Impact of AI Coding Assistants on Software Engineering: A Longitudinal Study». 2026.

[43] M. Fowler, «Design-First Collaboration». [En línea]. Disponible en: https://www.martinfowler.com/articles/reduce-friction-ai/design-first-collaboration.html

[44] IBM, «What Is Data Science?». [En línea]. Disponible en: https://www.ibm.com/think/topics/data-science

[45] IBM, «What Is Machine Learning?». [En línea]. Disponible en: https://www.ibm.com/think/topics/machine-learning

[46] IBM, «DevOps Lifecycle». [En línea]. Disponible en: https://www.ibm.com/think/topics/devops-lifecycle

[47] Google Cloud, «MLOps: Continuous Delivery and Automation Pipelines in Machine Learning». [En línea]. Disponible en: https://docs.cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

[48] Suhendar y Y. Widyani, «Machine Learning Application Development Guidelines Using CRISP-DM and Scrum Concept», en *15th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 2023.

[49] U. Fayyad, G. Piatetsky-Shapiro, y P. Smyth, «From Data Mining to Knowledge Discovery in Databases», *AI Magazine*, vol. 17, n.° 3, pp. 37-54, 1996, doi: 10.1609/aimag.v17i3.1230.

[50] A. Azevedo y M. F. Santos, «KDD, SEMMA and CRISP-DM: A Parallel Overview», en *Proceedings of the IADIS European Conference on Data Mining 2008*, 2008, pp. 182-185.

[51] H. Mason y C. Wiggins, «A Taxonomy of Data Science». [En línea]. Disponible en: http://www.dataists.com/2010/09/a-taxonomy-of-data-science/

[52] G. Piatetsky, «CRISP-DM, still the top methodology for analytics, data mining, or data science projects». [En línea]. Disponible en: https://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html

[53] G. Mariscal, Ó. Marbán, y C. Fernández, «A survey of data mining and knowledge discovery process models and methodologies», *The Knowledge Engineering Review*, vol. 25, n.° 2, pp. 137-166, 2010, doi: 10.1017/S0269888910000032.

[54] F. Martínez-Plumed *et al.*, «CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories», *IEEE Transactions on Knowledge and Data Engineering*, 2019, doi: 10.1109/TKDE.2019.2962680.

[55] S. Studer *et al.*, «Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology». 2020.

[56] Congreso Nacional de Chile, «Ley 21.719: Regula la Protección y el Tratamiento de los Datos Personales y Crea la Agencia de Protección de Datos Personales». 2024.

[57] Real Academia Española, «Adecuar». [En línea]. Disponible en: https://dle.rae.es/adecuar

[58] E. D. Moya Cisternas, «Implementación de una PMO en Ambiente Académico para Proyectos Tipo Emprendimiento Temprano: Caso Taller de Desarrollo de Software», Memoria de titulación para optar al título de Ingeniera Civil Informática, Santiago, Chile, 2017.

[59] Universidad Técnica Federico Santa María, «3IE: Instituto Internacional para la Innovación Empresarial». [En línea]. Disponible en: https://3ie.usm.cl/

[60] J. Escobar-Pérez y Á. Cuervo-Martínez, «Validez de contenido y juicio de expertos: una aproximación a su utilización», *Avances en Medición*, vol. 6, n.° 1, pp. 27-36, 2008.

[61] S. Jamieson, «Likert scales: how to (ab)use them», *Medical Education*, vol. 38, n.° 12, pp. 1217-1218, 2004, doi: 10.1111/j.1365-2929.2004.02012.x.