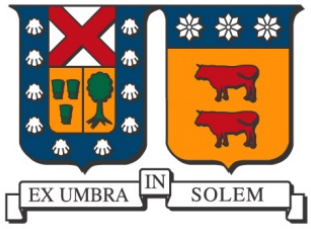
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO - CHILE



“DESCUBRIENDO LA PRÁCTICA REFLEXIVA DE LOS ESTUDIANTES EN PROYECTOS CAPSTONE DE SOFTWARE MEDIANTE EL ANÁLISIS INTELIGENTE DE SUS RETROSPECTIVAS DE SPRINT”

MARCOS MANUEL MALDONADO MERINO

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS DE LA INGENIERÍA INFORMÁTICA

Profesor Guía: Dra. Lioubov Dombrovskaia
Profesor Correferente Interno: Dr. Ricardo Ñanculef
Profesor Correferente Externo: Dr. Marcelo Mendoza

Marzo - 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: DESCUBRIENDO LA PRÁCTICA REFLEXIVA DE LOS ESTUDIANTES EN PROYECTOS CAPSTONE DE SOFTWARE MEDIANTE EL ANÁLISIS INTELIGENTE DE SUS RETROSPECTIVAS DE SPRINT

Nombre del candidato(a): Marcos Manuel Maldonado Merino

Carrera / Grado: Magíster en Ciencias de la Ingeniería Informática

Campus: Casa Central Valparaiso **;** **Departamento**: Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Lioubov Dombrovskaia, 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

☒ 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:

☐ 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: 20/03/2026 ; **Firma:** _____

Estudiante o Candidato(a):

Fecha: 20/03/2026 ; **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

Dedico este trabajo y todo el esfuerzo que le dediqué a este programa de posgrado a quienes vivieron de cerca este proceso conmigo, a quienes me vieron esforzarme cada día y a veces pusieron su mano sobre el timón.

Hicieron con su presencia y apoyo esto posible.

Gracias de corazón.

AGRADECIMIENTOS

En este trabajo que culmina mi programa de posgrado, quiero agradecer a mi profesora guía, la Dra. Lioubov Dombrovskaia, quien me dio apoyo e indicaciones fundamentales para lograr un trabajo de investigación excelente, las cuales me llevaron a cumplir con éxito todas las exigencias del programa de posgrado.

Agradezco al Dr. Ricardo Ñanculef y al Dr. Marcelo Mendoza, quienes fueron parte de la comisión revisora de este trabajo y me entregaron retroalimentación muy valiosa para mi estudio y también para mi vida profesional. Personalmente destaco la formación que me brindó el profesor Ricardo Ñanculef, quien además de ser parte de mi comisión revisora fue parte fundamental de mi formación como investigador, aprendizaje que valoraré y llevaré conmigo siempre, al igual que la grata experiencia en el programa junto él.

Agradezco la labor de los funcionarios y académicos del programa de posgrado, en especial a José Herrera quien, aquí en las dependencias de la universidad en Santiago, nos apoyó a los estudiantes de posgrado con cualquier duda o requerimiento de la sala de posgrado e investigación facilitada por la Dirección del Programa y de la cual hice uso por mucho tiempo. Sin duda los recursos facilitados por la Dirección del Programa, en especial la sala de investigación dotada con todas las herramientas necesarias para los estudiantes, facilitaron el trabajo de todos.

Y finalmente le expreso mi gratitud a todos mis cercanos que me entregaron contención, alegría y ayuda en este proceso. Gracias a mi familia, mis amigos, conocidos y las personas que estuvieron conmigo en los momentos de alegría y tristeza. Me ayudaron a superar con creces este desafío.

RESUMEN

Resumen— Los proyectos *capstone* simulan entornos profesionales ágiles donde la Retrospectiva de Sprint es fundamental para la mejora continua. No obstante, el contenido de estas reflexiones suele ser una “caja negra” para el diagnóstico docente masivo. Esta tesis propone una metodología híbrida de análisis inteligente para caracterizar la práctica reflexiva estudiantil, combinando un enfoque deductivo mediante Grandes Modelos de Lenguaje (LLMs) y uno inductivo con modelado de tópicos (usando BERTopic). Se procesó un corpus de 2219 frases provenientes de 156 retrospectivas de tres cohortes (2022-2024) de un curso *capstone* universitario. Los hallazgos revelan un predominio de la dimensión operativa. Se identificó una “Fricción Operativa Crónica”, donde problemas de planificación y ritmo persisten transversalmente. En contraste, destaca una fuerte “Resiliencia Social”, actuando las dinámicas humanas como soporte ante la frustración. Además, se detectó un “Estancamiento Reflexivo”, evidenciado por la disminución de la diversidad temática al final del proyecto. También se observó una visibilidad tardía de la deuda técnica.

Palabras Clave— Educación en Ingeniería de Software, Proyectos Capstone, Retrospectivas de Sprint, Procesamiento de Lenguaje Natural

ABSTRACT

***Abstract*—** Projects simulate agile professional environments where the Sprint Retrospective is fundamental for continuous improvement. However, the content of these reflections is often a ”black box”for large-scale teacher assessments. This thesis proposes a hybrid methodology for intelligent analysis to characterize students’ reflective practice, combining a deductive approach using Large Language Models (LLMs) and an inductive approach with topic modeling (using BERTopic). A corpus of 2219 sentences from 156 retrospectives across three cohorts (2022-2024) of a university course was processed. The findings reveal a predominance of the operational dimension. A “Chronic Operational Friction” was identified, where planning and pacing problems persist across the board. In contrast, a strong “Social Resilience” stands out, with human dynamics acting as a support in the face of frustration. Furthermore, a “Reflective Stagnation” was detected, evidenced by the decrease in thematic diversity at the end of the project. A delayed visibility of the technical debt was also observed.

***Keywords*—** Software Engineering Education, Capstone Projects, Sprint Retrospectives, Natural Language Processing

GLOSARIO

H.U.: Historia de Usuario.
HH.UU.: Historias de Usuario.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1
INTRODUCCIÓN

1.1. Contexto

La Ingeniería de Software se define como la disciplina que busca "la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, la aplicación de la ingeniería al software"[IEE, 1991]. Esta disciplina no solo requiere del conocimiento técnico de la construcción de sistemas de software, sino que también, dada su naturaleza socio-técnica [Storey *et al.*, 2020], requiere de la coordinación de las personas involucradas. En este sentido, el Desarrollo de Software Ágil (DSA) se ha instalado con fuerza dentro de la práctica ingenieril en la industria [Hoda *et al.*, 2018] y la academia [Kato y Van Greunen, 2023] como una aproximación al desarrollo de software más ligera que la aproximación tradicional [Aitken e Ilango, 2013].

Respecto a la enseñanza de la disciplina, los proyectos *capstone* de desarrollo de software en programas universitarios buscan simular el proceso de desarrollo de software lo más fiel a la práctica profesional posible, comúnmente enfatizando la adopción de los principios de DSA, i.e. principios ágiles. De entre los 12 principios ágiles establecidos en el "Manifiesto Ágil" [Beck *et al.*, 2001], la reflexión a intervalos regulares de tiempo sobre el proceso de desarrollo del equipo se vuelve un principio importante en la enseñanza de los estudiantes, ya que enfatiza el pensamiento crítico sobre las fortalezas y debilidades del equipo y refuerza la autonomía y la autogestión del mismo.

En Scrum [Schwaber y Sutherland, 2013], el marco de trabajo ágil más popular en estos proyectos, la reflexión a intervalos recurrentes se lleva a cabo a través de la ceremonia de *Retrospectiva de Sprint*, i.e. retrospectiva, la cual, en esencia, requiere que el equipo se reúna a discutir sobre su rendimiento reciente, definiendo también las acciones futuras a las que se comprometen para mejorar su proceso de desarrollo.

El enfoque de la investigación sobre la retrospectiva en Scrum se ha centrado principalmente en describir, proponer y validar nuevas formas de llevarla a cabo [Ozoliņš, 2018], especialmente a través de juegos [Przybyłek y Kotecka, 2017, Ng *et al.*, 2020, Przybyłek *et al.*, 2022, Östman y Hallmén, 2023, Matthies y Dobrigkeit, 2019, Marshburn y Sieck, 2019] y en el uso de análisis y datos históricos para impulsar su efecto positivo [Ozoliņš, 2018, Matthies, 2020, Sandoval-Alfaro y Quintero-Meza, 2021].

1.2. Desafío y Oportunidad

Hasta ahora, se sabe poco sobre el proceso de reflexión de los equipos de estudiantes en estas ceremonias, perdiendo conocimiento sobre la riqueza de su discusión y sobre los problemas y éxitos que los estudiantes identifican en su proceso de desarrollo, el cual es valioso para entender cómo realizan esta práctica y cómo a partir de esta los equipos ajustan sus prácticas de trabajo, finalmente descubriendo la utilidad que ellos obtienen de esta práctica.

Entonces, como primer paso para esto, el caracterizar el contenido de esta reflexión en el contexto académico, aprovechando la naturaleza iterativa de Scrum, se convierte en una valiosa herramienta para entender el comportamiento de los estudiantes a lo largo del desarrollo de los proyectos de DSA. Además, esto es importante para que profesores e instructores puedan hacer un diagnóstico fiel de la situación de los estudiantes y así tomar acciones correctivas en el transcurso o el diseño del proyecto o las materias del programa universitario previas a este.

Sin embargo, analizar los documentos que se pueden producir en este contexto, como resúmenes, informes o reportes, puede representar un desafío considerable si se utilizan técnicas tradicionales de análisis cualitativo. Estas consideran un proceso manual intensivo en tiempo, difícil de escalar (especialmente si se consideran múltiples equipos de estudiantes a lo largo de varios Sprints) y susceptible a la subjetividad y posible inconsistencia de los expertos que analizan manualmente los documentos. Para superar estas barreras y aprovechar la oportunidad de extraer valor de manera sistemática y oportuna, una gran oportunidad surge en la aplicación de métodos inteligentes de análisis de texto. Los avances en el Procesamiento del Lenguaje Natural (NLP) y la minería de texto ofrecen un camino para automatizar partes importantes de la caracterización de este contenido. Técnicas que van desde la modelización de tópicos hasta el análisis de sentimientos y la clasificación automática de texto, permiten procesar grandes volúmenes de discusión no estructurada, identificar patrones recurrentes, y, finalmente para este propósito, enmarcar con mayor claridad la naturaleza de los problemas y éxitos reportados por los estudiantes, transformando así esta rica retroalimentación en conocimiento accionable. Es por esto que es preciso explorar estos métodos para así facilitar la extracción de valor de estos documentos, facilitando también su utilidad tanto para los estudiantes como los instructores del curso.

1.3. Propuesta y Estructura

En este contexto, el presente trabajo de tesis aplica estos métodos a un caso de estudio concreto, el cual se estructura en las siguientes secciones: en el Capítulo 2 se establece el marco conceptual para facilitar el entendimiento de los conceptos más complejos y específicos al contexto del trabajo que son referidos y que fueron utilizados para el desarrollo de la tesis; luego, en el Capítulo 3 se discute la literatura relacionada a este trabajo y como este último

se enmarca y se diferencia de ellos; posteriormente, en el Capítulo 4 se describe el estudio de caso que se llevó a cabo para realizar el análisis inteligente sobre las Retrospectivas de Sprint, mostrando y discutiendo sobre los resultados de este en el Capítulo 5; finalmente, se presentan las conclusiones y se resumen los hallazgos más importantes del trabajo en el Capítulo 6.

CAPÍTULO 2
MARCO CONCEPTUAL

2.1. Agilidad, Scrum y las Retrospectivas de Sprint

La Agilidad en el desarrollo de proyectos de software se entiende como un marco general de trabajo para enfrentar estos proyectos que surgió a finales del siglo XX como una forma de alejarse de los enfoques tradicionales basados en planificación estricta, como la metodología en cascada donde el proceso de desarrollo de software se modela como un enfoque sencillo y por fases secuenciales en línea recta [Adenowo y Adenowo, 2013]. El Manifiesto Ágil, propuesto por Beck y colaboradores como el producto de la recopilación de años de experiencia profesional, describe valores y principios para el proceso de desarrollo de software como una guía para la conducción de estos proyectos [Beck *et al.*, 2001].

A partir de estos valores y principios, surgieron varios marcos y metodologías de desarrollo de software que se difundieron rápidamente entre las organizaciones de software. Scrum, uno de los marcos ágiles más populares, utiliza un enfoque iterativo-incremental para abordar la incertidumbre de los requisitos del software, principalmente manteniendo constante interacción con el cliente y con las partes interesadas [Schwaber y Sutherland, 2013]. Cada iteración se denomina *Sprint* y tiene una duración de algunas semanas y un tamaño medido en *Puntos de Historia*, una medida de esfuerzo para las historias de usuario que describen los requerimientos del proyecto. En Scrum, después de cada Sprint, se entrega al cliente una nueva versión funcional del software (o incremento de software) en una ceremonia llamada *Revisión del Sprint* donde usualmente los clientes brindan retroalimentación al equipo de desarrollo sobre el producto de software [Schwaber y Sutherland, 2013].

Finalmente, en la ceremonia de *Retrospectiva del Sprint*, el equipo de desarrollo reflexiona sobre su desempeño y los ajustes que son necesarios en el proceso de desarrollo. En esta ceremonia, se suele crear un documento resumen que contiene las reflexiones de los equipos sobre lo que salió bien y lo que salió mal en el Sprint, y lo que acordaron repetir o cambiar en el siguiente Sprint. Dependiendo de la manera o los “modelos” de reflexión que se utilicen, variará la forma en que se estructura la reflexión, ahondando o evadiendo ciertos aspectos del proceso. Lo que sí es transversal en las retrospectivas es que se requiere tanto que el equipo realice una descripción crítica del proceso de desarrollo, como también que dé su opinión sobre la forma en que sucedieron las cosas, brindándole a esta ceremonia, por lo tanto, un carácter descriptivo de los aspectos que fueron relevantes en el resultado del Sprint, ya sea para bien o para mal, pero a través de los ojos de los integrantes del equipo.

2.2. Proyectos Capstone en programas de Ingeniería de Software

Los Proyectos Capstone (o proyectos de fin de carrera) en Ingeniería de Software representan una experiencia pedagógica culminante, diseñada para que los estudiantes integren y apliquen el conjunto de conocimientos teóricos y habilidades prácticas adquiridas durante su formación [Clear y Veling, 2012]. Estos proyectos buscan simular un entorno de desarrollo profesional, cerrando la brecha entre la academia y la industria. Generalmente, involucran a equipos de estudiantes que deben gestionar el ciclo de vida completo de un producto de software—desde la conceptualización y levantamiento de requisitos hasta el diseño, implementación, pruebas y despliegue—para un cliente real o simulado. El objetivo es enfrentar a los estudiantes a un problema de una complejidad significativa, que no puede ser resuelto en un curso regular, forzándolos a tomar decisiones de diseño y arquitectura con consecuencias reales en el producto final.

Dada la naturaleza inherentemente incierta y la complejidad de estos proyectos, la literatura especializada reporta una fuerte inclinación hacia la adopción de metodologías ágiles como estrategia principal de gestión [Mahnic, 2012]. A diferencia de los modelos predictivos tradicionales (como el modelo en cascada), los marcos de trabajo ágiles—principalmente Scrum y Kanban—ofrecen la flexibilidad necesaria para que los equipos de estudiantes gestionen requisitos ambiguos o cambiantes. La adopción de estas prácticas no solo facilita la gestión del proyecto, sino que también entrena a los estudiantes en las dinámicas de colaboración y adaptación que predominan en la industria del software actual.

Si bien los proyectos capstone ofrecen oportunidades valiosas para el desarrollo de competencias técnicas y habilidades blandas—tales como la comunicación, la negociación y la resolución de conflictos [Robles *et al.*, 2017]—también exponen a los estudiantes a desafíos considerables. Los problemas más comúnmente reportados en la literatura incluyen la gestión de la dinámica de equipo (conflictos interpersonales, contribución desigual), la definición y control del alcance, y dificultades en la integración técnica [Kifetew *et al.*, 2018]. Estas dificultades subrayan la necesidad crítica de mecanismos estructurados de reflexión y mejora continua. Es en este contexto de alta presión, aprendizaje práctico y problemas de proceso donde las prácticas de reflexión, como las retrospectivas de Sprint, adquieren una relevancia fundamental para el éxito del aprendizaje y del proyecto.

2.3. Análisis inteligente sobre texto

El análisis inteligente de textos es un campo multidisciplinar que emplea técnicas avanzadas de procesamiento de lenguaje natural (PLN) y aprendizaje automático para extraer conocimiento, patrones y temáticas a partir de grandes corpus de datos no estructurados. Entre las tareas más populares se incluyen el modelado de tópicos, la clasificación de textos, la detección de sentimientos, el análisis de discurso y la anotación automática de información relevante. Estas tareas permiten transformar textos complejos y exten-

sos en información procesable, tal como se demuestra en investigaciones orientadas al análisis de foros médicos, textos académicos o datos empresariales. La integración de inteligencia artificial potencia estas aplicaciones, logrando mayor precisión y rapidez, incluso en aspectos subjetivos como la ironía, la intencionalidad y el contexto en el discurso [Herhausen *et al.*, 2025, Törnberg, 2023, Castellanos *et al.*, 2025].

2.4. Grandes Modelos de Lenguaje

Un Gran Modelo de Lenguaje (LLM) es un modelo probabilístico cuyo objetivo es aprender una distribución de probabilidad sobre secuencias de texto. El conocimiento fundamental del modelo se adquiere mediante el entrenamiento sobre un corpus lingüístico: un conjunto de datos masivo y estructurado, compuesto por miles de millones de textos (como libros, artículos web y código fuente) que representan la totalidad del lenguaje que el modelo debe aprender. Para procesar esta información, el texto del corpus no se divide en palabras, sino en tokens. Un token es la unidad de procesamiento fundamental del modelo; puede ser una palabra completa (“casa”), una sub-palabra (como “auto-” y “móvil” en “automóvil”) o incluso un signo de puntuación. Esta segmentación permite al modelo manejar vocabulario desconocido y una morfología compleja.

Históricamente, los primeros intentos estadísticos para modelar estas secuencias de tokens fueron los N-gramas [Kneser y Ney, 1995]. Para superar las limitaciones de contexto fijo de estos, se introdujeron los modelos basados en Redes Neuronales. Una red neuronal es un modelo computacional compuesto por capas de nodos interconectados con pesos ajustables. El aprendizaje ocurre durante el entrenamiento mediante retropropagación (backpropagation), un proceso donde el modelo ajusta sus pesos para minimizar el error entre sus predicciones y los datos reales del corpus.

El hito que define potencia el uso de redes neuronales para el aprendizaje del lenguaje humano fue el de la propuesta de la arquitectura Transformer [Vaswani *et al.*, 2017]. Su innovación central por sobre las redes tradicionales es el mecanismo de autoatención (self-attention), que permite al modelo procesar todos los tokens de la secuencia de forma simultánea (paralela). Conceptualmente, para cada token de entrada, el mecanismo de autoatención genera tres vectores: una Query (Q) (que representa lo que el token está “buscando”), una Key (K) (que representa lo que el token “es” o su función) y un Value (V) (que representa el contenido o significado del token). El modelo calcula una puntuación de afinidad comparando el vector Query de un token con los vectores Key de todos los demás tokens en la secuencia. Estas puntuaciones, tras normalizarse, determinan cuánta “atención” debe prestar cada token a los demás, y se usan para crear una suma ponderada de todos los vectores Value. El resultado es una nueva representación para cada token que está profundamente contextualizada por su relación con toda la secuencia, capturando dependencias complejas sin importar la distancia de cada token dentro de la secuencia.

Las capacidades de los LLM modernos se manifiestan de dos formas principales. La primera

es la generación de texto, que opera bajo un proceso autorregresivo [Brown *et al.*, 2020]. Esto describe un procedimiento iterativo, token por token, donde el modelo predice el siguiente token basándose en la secuencia de entrada y en todos los tokens que ya ha generado. Esto, sumado a la gran escala de estos modelos (billones de tokens y miles de millones de parámetros) ha dado lugar a una capacidad emergente conocida como aprendizaje en contexto (in-context learning o ICL). El ICL es la habilidad del modelo para realizar tareas nuevas sin un re-entrenamiento explícito. A diferencia del ajuste fino —que es un segundo proceso de entrenamiento donde el modelo se especializa en una tarea con datos etiquetados—, el ICL ocurre en tiempo de inferencia. Simplemente proporcionando al modelo un prompt o "instrucciones" que contenga ejemplos de la tarea (few-shot) o solo su descripción (zero-shot), el modelo reconoce el patrón y lo replica.

La segunda capacidad fundamental, que subyace a todas las demás, es la generación de embeddings. Un embedding es una representación vectorial densa —una lista de números de alta dimensionalidad— que captura el significado semántico y contextual de un token o de una secuencia de texto. Estos vectores posicionan el significado en un espacio matemático, donde conceptos semánticamente similares (ej. "rey" y "reina") ocupan posiciones relativas coherentes. Estos embeddings son la materia prima para tareas como la clasificación de texto (ej. análisis de sentimientos).

2.5. Modelado de Tópicos en texto y BERTopic

El modelado de tópicos es una técnica fundamental en el PLN que permite descubrir estructuras temáticas latentes dentro de grandes colecciones de documentos no estructurados. Su objetivo es identificar patrones de palabras que co-ocurren con frecuencia, agrupándolas para representar un "tópico" o concepto abstracto. A modo de ilustración, si analizamos miles de reseñas de un producto tecnológico, un modelo de tópicos eficaz debería ser capaz de separar automáticamente comentarios sobre "batería", "carga" y "duración" en un grupo (Tópico A: Energía), y comentarios sobre "pantalla", "brillo" y "resolución" en otro (Tópico B: Visualización), sin necesidad de etiquetado previo por parte de humanos.

Tradicionalmente, este problema se ha abordado mediante métodos probabilísticos y de factorización de matrices, siendo la Asignación Latente de Dirichlet (LDA) y la Factorización de Matrices No Negativas (NMF) los estándares de la industria. LDA, por ejemplo, asume que cada documento es una mezcla probabilística de varios tópicos y que cada tópico es una mezcla de palabras. Sin embargo, estos modelos clásicos suelen basarse en un enfoque de "bolsa de palabras" (bag-of-words), lo que significa que ignoran el orden y el contexto semántico de las palabras, limitando su capacidad para distinguir significados en frases complejas o detectar la polisemia (palabras con múltiples significados según el contexto).

Para evaluar la eficacia de estos modelos frente a las nuevas generaciones, se utilizan métricas clave como la coherencia semántica y la diversidad temática. La coherencia mide el grado en que las palabras principales de un tópico están relacionadas semánticamente entre sí (es

decir, si tienen sentido juntas para un humano), mientras que la diversidad evalúa qué tan diferentes son los tópicos entre sí, evitando la redundancia. Los modelos tradicionales a menudo sacrifican una métrica por la otra; un modelo puede generar tópicos muy coherentes pero repetitivos, o muy diversos pero con palabras sin relación lógica.

En este escenario surge BERTopic, una técnica que redefine el estado del arte aprovechando la potencia de los transformers. Según su artículo fundacional [Grootendorst, 2022], BERTopic no trata los documentos como bolsas de palabras, sino que genera embeddings (representaciones vectoriales densas) de cada documento utilizando modelos de lenguaje preentrenados para capturar su contexto semántico. Su arquitectura opera en tres etapas: primero reduce la dimensionalidad de los embeddings (usualmente con UMAP), luego agrupa los documentos semánticamente similares (mediante HDBSCAN) y, finalmente, extrae las palabras clave representativas de cada clúster utilizando una variante de TF-IDF basada en clases, denominada c-TF-IDF. Esta metodología permite a BERTopic superar a LDA y NMF en coherencia y diversidad, al entender que “banco” en un contexto financiero es distinto a “banco” en un contexto de mobiliario [Ma *et al.*, 2025, Khodeir y Elghannam, 2025].

Finalmente, la evolución de este campo no se detiene en la agrupación. La integración de BERTopic con LLMs, como GPT-4 o LLaMA, permite dar el siguiente paso en la generación de mejores representaciones latentes para los documentos que, dada la riqueza y variedad del corpus en los que fueron entrenados estos modelos, aporten en última instancia a una generación de tópicos más fina y representativa, ampliando incluso su aplicación en dominios de alta especialidad médica o técnica [Xie *et al.*, 2025, Törnberg, 2023, Castellanos *et al.*, 2025].

CAPÍTULO 3
LITERATURA RELACIONADA

3.1. Temáticas y Dinámicas de Discusión en Retrospectivas

El contenido de las retrospectivas varía significativamente dependiendo del nivel de madurez del equipo y el entorno en el que se desarrollan. En el ámbito industrial, la literatura describe estas reuniones como espacios complejos de regulación social y técnica. Estudios longitudinales en grandes organizaciones, como los de [Lehtinen *et al.*, 2017] y [Dingsøyr *et al.*, 2018], indican que los profesionales dedican la mayor parte del tiempo a discutir la eficiencia del proceso de desarrollo y las herramientas de implementación. Sin embargo, un hallazgo crucial de estos trabajos es que las discusiones sobre relaciones interpersonales y "temas blandos" son recurrentes y persistentes en el tiempo, sugiriendo que los equipos ágiles utilizan estos espacios no solo para corregir el rumbo técnico, sino para mantener la cohesión grupal.

Profundizando en la dimensión humana, [Andriyani *et al.*, 2017] observaron que los equipos industriales van más allá de la mera identificación de obstáculos; se involucran en debates sobre sentimientos y razones subyacentes. Esta capacidad de reflexión profunda permite a los profesionales generar planes de acción concretos para futuros Sprints. No obstante, la investigación también advierte que no todos los equipos logran este nivel de introspección, existiendo casos donde se evita deliberadamente hablar de emociones para centrarse exclusivamente en el rendimiento pasado, lo que limita el potencial de mejora continua.

En contraste, al trasladar el foco a los entornos académicos, la naturaleza de la discusión cambia hacia necesidades más inmediatas. [Gestwicki y McNely, 2013] encontraron que, en equipos de estudiantes, casi la mitad de las discusiones se clasifican como "colaboración". A diferencia de los expertos que refinan procesos, los estudiantes utilizan la retrospectiva principalmente como un mecanismo de supervivencia pragmática para coordinar la construcción del sistema y resolver la cohesión social básica. Esto sugiere que, en etapas formativas, la retrospectiva actúa más como una reunión de sincronización que como una herramienta de mejora de procesos de alto nivel.

Más recientemente, [Hundhausen *et al.*, 2024] aportaron evidencia crítica sobre la calidad de estas reflexiones en proyectos universitarios. Aunque los estudiantes logran identificar problemas en sus prácticas de trabajo y comunicación, sus reportes tienden a ser descriptivos en lugar de analíticos. Los equipos frecuentemente proponen estrategias prácticas sin una justificación clara de por qué esas acciones resolverán el problema raíz. Esta falta de profundidad reflexiva marca una diferencia fundamental con la industria y plantea el desafío pedagógico de cómo guiar a los estudiantes desde la descripción de eventos hacia el análisis de causas.

A pesar de que estos trabajos permiten categorizar los temas generales (comunicación, tra-

bajo, aprendizaje), existe un vacío en la literatura respecto al análisis granular y longitudinal de estas discusiones mediante métodos automáticos. La mayoría de los estudios actuales se basan en codificación manual o análisis estáticos, lo que impide caracterizar con precisión cómo evolucionan las problemáticas específicas semana a semana. Esta tesis aborda dicha limitación, proponiendo el uso de modelos de lenguaje para clasificar y rastrear la "temperatura" de estos tópicos a lo largo de todo el ciclo de vida de un proyecto capstone.

3.2. Desafíos Comunes en Proyectos Capstone de Software

La literatura especializada coincide en que, paradojalmente, la ingeniería de software es a menudo "la parte fácil" de los proyectos de final de carrera. Un estudio reciente de [Li *et al.*, 2023], que sintetiza lecciones de cursos capstone a gran escala, sugiere que mientras los estudiantes suelen poseer las competencias técnicas para construir el producto, carecen de las herramientas para manejar la ambigüedad y la dinámica social del desarrollo. Este hallazgo resuena con investigaciones longitudinales previas, como las de [Vanhanen y Lehtinen, 2014] en la Universidad Aalto, quienes identificaron que los problemas de comunicación y coordinación superan con creces a los desafíos puramente técnicos de funcionalidad o calidad del código.

Profundizando en la naturaleza de estos obstáculos, [Sedelmaier y Landes, 2020] categorizan las dificultades en tres dimensiones críticas: humana, organizacional y profesional. Su análisis cualitativo revela que la "fricción comunicacional" no es solo interna, sino que se exacerba en la interacción con los stakeholders. Los estudiantes, habituados a entornos académicos controlados, luchan por gestionar las expectativas cambiantes de los clientes y la complejidad inherente de los requisitos del mundo real, lo que a menudo deriva en una crisis de liderazgo dentro de los equipos al no saber cómo repartir responsabilidades de manera equitativa.

Un factor determinante en esta problemática es la transición abrupta desde un aprendizaje dirigido por el instructor hacia uno autodirigido. [Majanoja y Vasankari, 2018] observan que los estudiantes experimentan dificultades significativas al asumir la propiedad de la gestión del proyecto. Esta falta de experiencia en autogestión se manifiesta en estimaciones de esfuerzo poco realistas y una planificación deficiente, lo que inevitablemente conduce al incumplimiento de plazos. La ingeniería de requisitos, en particular, se convierte en un punto de dolor recurrente, ya que los equipos novatos tienden a subestimar la volatilidad de las necesidades del cliente.

En el contexto específico de la educación en ingeniería en Chile, [Bastarrica *et al.*, 2017] aportan una visión crucial sobre la autopercepción de los estudiantes. Antes de enfrentar el curso capstone, los alumnos tienden a subestimar la complejidad del trabajo en equipo, confiando excesivamente en sus habilidades técnicas individuales. Es solo a través del choque con la realidad del proyecto que reconocen la importancia vital de las habilidades blandas. Este "golpe de realidad" sugiere que los problemas no surgen por falta de capacidad, sino por una disonancia inicial entre las expectativas académicas y las demandas de un entorno pro-

fesional simulado.

A pesar de que las metodologías ágiles como Scrum están diseñadas para mitigar estos riesgos a través de la inspección y adaptación continua, la literatura indica que la adopción de estas prácticas es, en sí misma, un desafío. Los estudiantes a menudo realizan las ceremonias ágiles de manera mecánica sin lograr una reflexión profunda sobre sus procesos de trabajo. Si bien reconocen al final del curso un crecimiento en su identidad profesional y confianza [Lutz y Paretti, 2017], el proceso intermedio sigue siendo una "caja negra" de conflictos y aprendizajes desestructurados.

Por consiguiente, existe una necesidad latente de monitorear cómo estos problemas evolucionan semana a semana. Aunque se han documentado los problemas post-mortem, la literatura carece de estudios que utilicen el análisis inteligente de los artefactos generados durante el proceso —específicamente las reflexiones en las retrospectivas— para detectar estos patrones de comportamiento en tiempo real. Esta tesis busca precisamente iluminar esa área, utilizando el texto de las retrospectivas como una ventana para observar y clasificar estas dificultades recurrentes.

3.3. ¿Qué aplicaciones del Análisis Inteligente se han registrado para el análisis de artefactos en Ingeniería de Software?

El análisis automático de información textual ha sido ampliamente explorado en la Ingeniería de Software para asistir en la gestión y mejora de procesos. Un área consolidada es el tratamiento de reportes de incidencias (*bug reports*) y comentarios de usuarios (*app reviews*). Investigaciones como las de [Maalej y Nabil, 2015] han utilizado técnicas de procesamiento de lenguaje natural para clasificar automáticamente comentarios de usuarios en categorías útiles para los desarrolladores, tales como reportes de errores, solicitudes de funcionalidades o simples opiniones. Este tipo de trabajos demuestra la utilidad de transformar texto no estructurado en información accionable para la toma de decisiones técnicas, un objetivo análogo al perseguido en el análisis de retrospectivas.

En un ámbito más cercano a la gestión de requisitos, el análisis inteligente se ha empleado para detectar y clasificar Requisitos No Funcionales (NFR) dentro de especificaciones de software. Estudios como los de [Kurtanović y Maalej, 2017] han aplicado técnicas de aprendizaje supervisado para identificar automáticamente si una frase en un documento de requisitos se refiere a seguridad, usabilidad o rendimiento. En su metodología, se propone segmentar y etiquetar frases específicas dentro de un documento más amplio, aportando una valiosa perspectiva para el análisis del contenido textual más grande a nivel de sentencias individuales.

Por otro lado, el análisis de los canales de comunicación humana dentro de los equipos de desarrollo ha cobrado gran relevancia. Se han realizado estudios sobre los historiales de chat (como Slack o IRC) y listas de correo para entender las emociones y dinámicas sociales de

los desarrolladores. [Calefato *et al.*, 2018] demostraron que es posible minar estos canales para evaluar el "sentimiento" del equipo y cómo este correlaciona con la productividad o la resolución de problemas. Este tipo de análisis comparte con esta tesis el interés por indagar en los aspectos más humanos y de la salud del equipo que se puedan encontrar en estos artefactos, más allá de los aspectos puramente técnicos del código.

Específicamente en el contexto de las metodologías ágiles, el análisis de las retrospectivas de Sprint ha sido abordado para asistir a los lideres de equipo. [Matthies *et al.*, 2019] propusieron herramientas para analizar los puntos discutidos en las retrospectivas, buscando distinguir automáticamente entre problemas reportados y acciones de mejora propuestas. Estos antecedentes confirman que el texto generado en estas ceremonias posee una estructura latente que puede ser explotada computacionalmente para ofrecer retroalimentación sobre la calidad del proceso ágil.

Finalmente, la irrupción de los LLMs ha expandido las fronteras de estas aplicaciones hacia tareas generativas y de razonamiento complejo. Investigaciones recientes recopiladas por [Hou *et al.*, 2023] muestran cómo estas nuevas tecnologías se están utilizando no solo para clasificar, sino para resumir discusiones técnicas, generar casos de prueba a partir de descripciones textuales y detectar duplicidad en reportes con una precisión casi humana. Esto señala una tendencia hacia herramientas que no solo procesan estadísticas de palabras, sino que interpretan la semántica incluso de artefactos de software como el código fuente.

En síntesis, la literatura evidencia que la aplicación de inteligencia artificial sobre artefactos textuales es una práctica válida y valiosa en la ingeniería de software moderna. Existe un claro precedente de éxito al aplicar estas técnicas para estructurar información caótica (como reviews o chats) y categorizarla según taxonomías predefinidas. La presente tesis se inserta en este ecosistema, aprovechando la madurez de estas aplicaciones para abordar un contexto educativo específico: la evaluación y seguimiento de equipos de estudiantes en proyectos capstone, donde la información cualitativa suele poner a prueba la capacidad de revisión manual de los docentes y, más importante, suele ser desaprovechada por profesores y alumnos como una fuente de información accionable. Existe espacio para nuevas oportunidades o enfoques en este sentido.

CAPÍTULO 4
ESTUDIO DE CASO

4.1. Propuesta y Contexto del Caso

Teniendo en cuenta el marco teórico descrito en la sección anterior, estudiar el contenido de la reflexión iterativa que se presenta en el caso de las retrospectivas de Scrum apoyándose de técnicas de análisis inteligente se convierte, entonces, en una oportunidad novedosa para conocer de mejor manera cómo los estudiantes describen y ajustan el proceso de desarrollo cuando enfrentan proyectos capstone de DSA. Por lo tanto, en la presente tesis, se contribuye al área de educación en Ingeniería de Software estudiando el contenido de las Retrospectivas de Sprint de los proyectos capstone de software que se realizan en el programa de Ingeniería Civil Informática de la Universidad Técnica Federico Santa María en el Campus San Joaquín. El curso capstone es de último año, y se divide en las asignaturas semestrales “Gestión de proyectos informáticos” y “Taller de desarrollo de proyectos informáticos” (desde ahora ambas llamadas “Feria de Software”).

El proyecto capstone “Feria de Software” tiene dos propósitos académicos: 1) ayudar a los estudiantes a adquirir experiencia en el desarrollo; y la gestión de proyectos de software innovadores y emprendedores; y 2) a mejorar sus habilidades de trabajo en equipo. Por lo tanto, si bien los estudiantes deben aplicar las habilidades duras para desarrollar una solución, también deben aplicar técnicas intensivas de gestión relacionadas con el trabajo en equipo, como la delegación de tareas y las metodologías de desarrollo de software.

La Figura 1 muestra una ilustración del proceso de desarrollo dividido por etapas numeradas. Cada proyecto está compuestos por equipos de seis estudiantes, habiendo en cada equipo un líder. En la primera etapa, los equipos identifican un problema de ingeniería complejo que resolver y buscan un experto o cliente que pueda aportar el conocimiento necesario del dominio en que se inserta. Es en esta etapa en donde los estudiantes definen los requerimientos a través de historias de usuario, estimando el esfuerzo que comprende cada una de ellas utilizando puntos de historia. Si bien cada proyecto es diferente, los instructores del curso revisan que los requerimientos de cada proyecto comprendan un esfuerzo requerido similar, correspondiente a 15 créditos SCT.

Respecto a la metodología de desarrollo, los instructores enfatizan Scrum y fomentan sus prácticas ágiles, como la realización de reuniones diarias, tableros Kanban y reuniones de Retrospectivas de Sprint, estás últimas siendo de carácter obligatorio. Los instructores y sus asistentes actúan como Scrum Masters en cada equipo, impartiendo clases magistrales y orientando a los equipos cuando tienen problemas o tienen preguntas sobre su proceso de desarrollo, el incremento de software o el rendimiento de los miembros del equipo, incluyendo aspectos académicos. Sin embargo, los estudiantes son los únicos responsables del progreso del proyecto y de la interacción con el cliente.

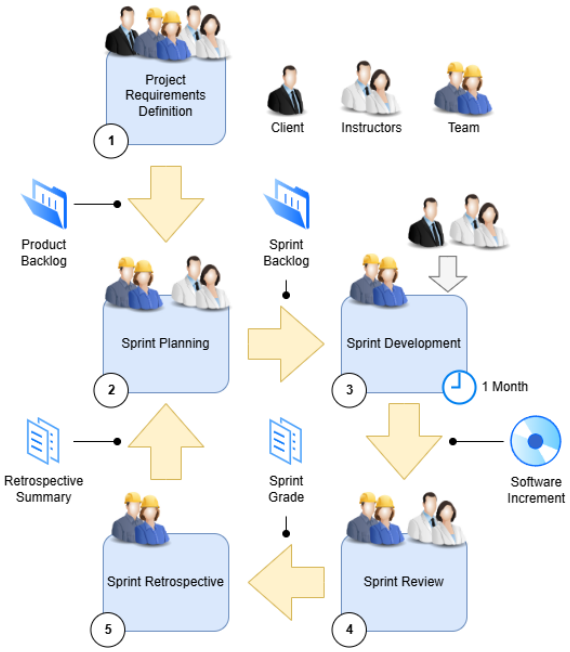


Figura 1: Artefactos y actores estrictamente (en contacto con cada recuadro) y no-estrictamente involucrados (fuera de cada recuadro) en cada fase.

Finalmente, respecto a las iteraciones del proceso de desarrollo, los equipos desarrollan su solución en cuatro Sprints de un mes de duración cada uno. Durante el primer Sprint, se incita a que las historias de usuario más críticas para la solución de software se incorporen bajo un producto mínimo viable (MVP). Las historias de usuario restantes se implementan en los tres Sprints siguientes, buscando siempre cumplir el mínimo de puntos de historia que se pide para cada Sprint.

La ceremonia de Revisión del Sprint es realizada por un jurado compuesto por los instructores y sus asistentes. Durante esta ceremonia, el incremento del software se compara con los criterios de aceptación de las historias de usuario desarrolladas. Es después de cada Revisión del Sprint que los equipos celebran la Retrospectiva del Sprint y crean un documento que resume su discusión. El modelo de Retrospectiva enfatizado por los instructores considera tres dimensiones de análisis para la iteración: 1) ¿Qué salió bien en el Sprint? 2) ¿Qué salió mal en el Sprint? y 3) ¿Qué se debe mantener en el siguiente Sprint? Si bien completar esta actividad es obligatorio, los equipos no son evaluados por la forma en que realizan la ceremonia ni por la calidad o riqueza de su reflexión, sino que sirven únicamente a los equipos.

En la presente tesis se estudiaron los resúmenes generados del proyecto por los equipos en los cohortes 2022 (15 equipos en el Sprint 1, 2 y 3), 2023 (15 equipos en el Sprint 0, 1, 2 y 3) y 2024 (17 equipos en el Sprint 1, 2 y 3), análizando un total de 156 resúmenes. Como consideración adicional, a diferencia de los años 2023 y 2024, en el año 2022 se instó un modelo de retrospectiva ligeramente diferente, buscando responder las preguntas: 1) ¿Qué se debería comenzar a hacer?; 2) ¿Qué se debería dejar de hacer?; y 3) ¿Qué se debería

mantener?.

4.2. Preguntas de investigación

Con la propuesta descrita anteriormente, se propone responder a las siguientes preguntas de investigación para el caso del proyecto “Feria de Software” en los cohortes indicados:

1. **¿Qué temas, con qué frecuencia y bajo que contexto son aludidos estos en las Retrospectivas de Sprint?** Con esto podremos descubrir la cobertura que tienen estos temas en la totalidad del contenido de las retrospectivas, conociendo si hay temas más o menos frecuentes o que simplemente jamás se aluden y finalmente para determinar si estos temas se mencionan en contextos particulares y con ciertas connotaciones de sentimiento (de manera positiva, neutral o negativa).
2. **¿Qué tan variada o rica es la reflexión de los equipos cuando realizan Retrospectivas de Sprint?** Tanto como es preciso conocer la frecuencia en que se aluden los temas clave, también es preciso conocer que tan rica es la reflexión de los equipos en términos de la cantidad de aspectos diferentes que son tomados en cuenta al momento de hacer las Retrospectivas de Sprint.
3. **¿Cómo evoluciona la reflexión de los equipos entre Sprints cuando realizan Retrospectivas de Sprint?** En la misma línea que la pregunta anterior, también es preciso conocer cómo los temas estudiados evolucionan a lo largo del transcurso del proyecto en los distintos equipos.
4. **¿Cuáles son los problemas y éxitos más importantes que se pueden extraer del análisis y en qué contexto aparecen dentro de las Retrospectivas de Sprint?** Cómo pregunta medular, es preciso determinar si los objetivos académicos del curso se están alcanzando o si se necesita tomar acción. La respuesta a esta pregunta permitirá obtener un diagnóstico fiel de la situación de los equipos cuando se enfrentan a este tipo de proyectos.

4.3. Metodología para la caracterización del contenido de las Retrospectivas de Sprint

Para responder las preguntas de investigación se buscará hacer una caracterización del contenido de los resúmenes utilizando un enfoque híbrido: usando una técnica de análisis deductivo para dirigir el análisis hacia un marco temático predefinido apoyándose de la literatura y que sea relevante para el interés de profesores y estudiantes, y también usando una técnica de análisis inductivo que permitan la aparición de patrones semánticos dentro del texto sin supervisión o ajuste dentro de un marco teórico, para así complementar

el análisis con una mirada enfocada a buscar patrones intrínsecos de contenido dentro de los resúmenes. Es por esto que, respecto a las técnicas a utilizar, en esta tesis se utilizan dos técnicas de análisis de inteligente sobre texto del estado del arte: Para el llevar a cabo el análisis deductivo se realiza una clasificación sobre texto especializado utilizando LLMs; y para el análisis inductivo se realiza un proceso de modelado de tópicos utilizando BERTopic [Grootendorst, 2022].

Con este análisis híbrido a través de dos tareas de NLP con técnicas del estado del arte se busca entonces:

- **Mitigar el sesgo de confirmación:** Al integrar un enfoque inductivo, se evita forzar la interpretación de los datos exclusivamente hacia las 23 categorías predefinidas, permitiendo identificar fenómenos emergentes, anomalías o temáticas específicas del contexto estudiantil que no están contempladas en la teoría general.
- **Lograr una Triangulación Metodológica:** La convergencia de resultados obtenidos mediante dos técnicas distintas incrementa la robustez y la validez interna del estudio, confirmando que los hallazgos no son artefactos de un único algoritmo.
- **Enriquecer la granularidad semántica:** Mientras que la clasificación con LLMs asigna etiquetas de alto nivel (el "*qué*" categoría), BERTopic desagrega el contenido semántico específico (el "*de qué*" se habla específicamente). Por ejemplo, una frase puede clasificarse deductivamente como "Tecnologías", pero inductivamente BERTopic podría revelar si se trata específicamente sobre "Git" o "Jira" y en qué contexto se menciona.
- **Capturar la "voz del equipo" sin filtros teóricos:** El enfoque deductivo estandariza el lenguaje hacia una taxonomía académica, mientras que el enfoque inductivo permite que el vocabulario, la jerga natural utilizada y la expresión de los estudiantes en sus proyectos capstone se considere en el análisis, ofreciendo una representación más fiel de la realidad fenomenológica de los equipos.

Esto permite construir sobre el trabajo previo realizado, el cuál ha sido presentado en la Conferencia Latinoamericana de Informática en su edición 2025 (CLEI 2025) dentro del track de Educación en Informática. En ese trabajo, se aplicó una técnica manual de análisis deductivo que permitió extraer las prácticas ágiles que habían sido reportadas como adoptadas por los estudiantes. En ese trabajo se detectó que el marco temático utilizado no era comprensivo de todo el contenido de los resúmenes, es decir, los temas escogidos solo aparecían en ciertas partes de los resúmenes, dejando la mayoría del contenido excluído de la obtención de conclusiones. Ahora, buscando una caracterización más completa de la reflexión de los estudiantes, se busca obtener de manera ágil una perspectiva con mayor riqueza y cobertura respecto del contenido de los documentos.

La organización general de la metodología que se llevó a cabo en esta tesis se puede encontrar en la Figura 2, donde se muestran los 3 puntos medulares para el procesamiento de los

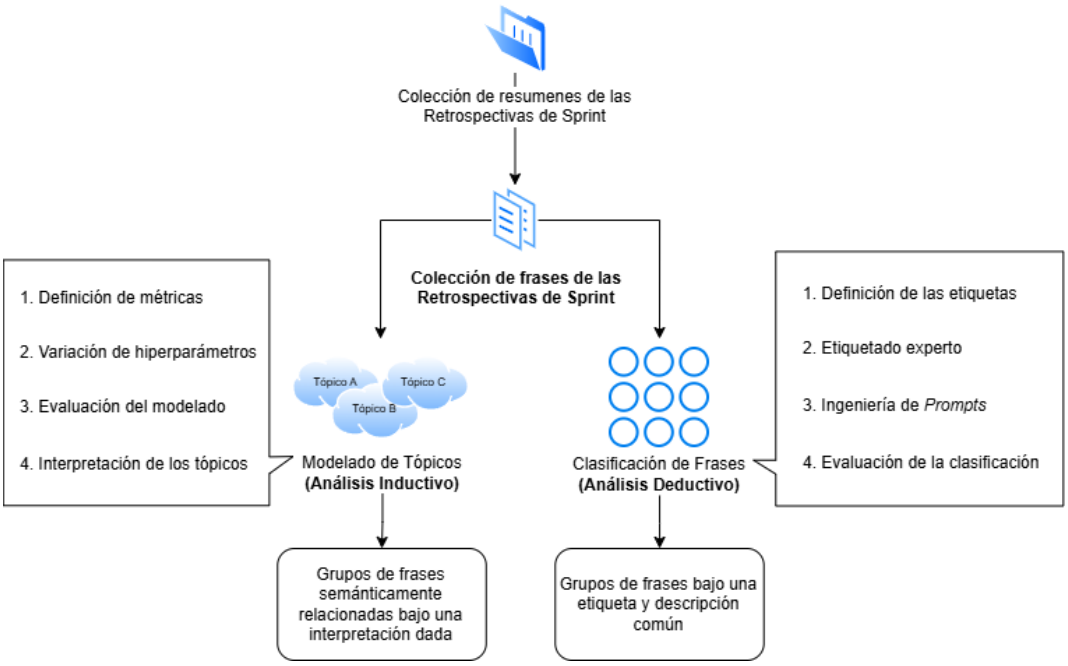


Figura 2: Organización general de la metodología de análisis híbrida.

resúmenes: la generación de una colección de frases que representen todo el contenido de las Retrospectivas de Sprint; la aplicación del modelado de tópicos usando esa colección de frases; y la clasificación de esa colección de frases dentro de etiquetas de interés.

Respecto al primer paso, para analizar las retrospectivas, estas se separaron por las frases que las componen (frases comprendidas entre dos puntos seguidos), teniendo así que cada frase pertenece a la retrospectiva respondiendo a una determinada pregunta (¿Qué salió bien?, ¿Qué salió mal?, etc.), perteneciendo a un equipo que realiza la reflexión en el contexto de un determinado Sprint. Esta separación granular de los resúmenes está motivada en mantener una aproximación similar para la aplicación de ambas técnicas: utilizar la clasificación de temas y el modelado de tópicos a nivel de frases para así también considerar la información de donde está ubicada y aislar la información de otras frases, teniendo una perspectiva más clara y segmentada del contenido de las retrospectiva. Finalmente, se quitó de todo el corpus de frases todas las referencias a los nombres o datos personales de los estudiantes, profesores o ayudantes del curso.

4.3.1. Modelado de Tópicos en las frases de las Retrospectivas de Sprint

Como se ilustra en la Figura 2, para llevar a cabo el modelado de tópicos sobre las frases de las retrospectivas utilizando la técnica del estado del arte BERTopic [Grootendorst, 2022], se ejecutó un diseño experimental exhaustivo que iteró sobre múltiples configuraciones, integrando la definición de métricas, la variación de hiperparámetros y la evaluación de resulta-

dos en un único flujo de trabajo.

En una primera instancia, para objetivar la calidad de los tópicos y seleccionar la mejor configuración, se estableció un marco de evaluación híbrido inspirado en la propuesta de [Pereira *et al.*, 2025], el cual combinó métricas de coherencia semántica con métricas de estructura matemática. Las métricas de calidad de tópicos se calcularon sobre las k palabras más representativas de cada tópico (con $k = 10$), las cuales se obtuvieron aplicando el método "*class-based TF-IDF*" (c-TF-IDF) [Grootendorst, 2022]. Este algoritmo pondera la importancia de un término dentro de un clúster específico frente a su frecuencia global, tal como se describe en la Ecuación 1:

$$W_{t,c} = tf_{t,c} \cdot \log\left(1 + \frac{A}{tf_t}\right) \tag{1}$$

Donde $W_{t,c}$ es el puntaje de relevancia del término t en la clase c. El término $tf_{t,c}$ representa la frecuencia de aparición del término t dentro de la clase c, mientras que A corresponde al promedio de palabras por clase. Finalmente, tf_t indica la frecuencia total del término t en todas las clases.

Basado en los términos con mayor puntaje $W_{t,c}$, se procedió al cálculo de cuatro métricas fundamentales. Primero, se utilizó la **NPMI (Normalized Pointwise Mutual Information)**, la cual mide la coherencia semántica cuantificando la probabilidad de co-ocurrencia de palabras [Bouma, 2009]. Su fórmula es:

$$\text{NPMI} = \frac{\log(\frac{P(w_1,w_2)}{P(w_1)P(w_2)})}{-\log(P(w_1,w_2))} \tag{2}$$

Donde $P(w_1, w_2)$ es la probabilidad conjunta de que las palabras w_1 y w_2 aparezcan juntas en un documento, mientras que $P(w_1)$ y $P(w_2)$ son las probabilidades marginales. Valores cercanos a 1 indican una alta coherencia. En segundo lugar, se calculó la **Coherencia TF-IDF**, que evalúa la calidad del tópico asegurando que las palabras frecuentes dentro del clúster no sean simplemente palabras comunes del corpus general [Nikolenko *et al.*, 2015]. Se define mediante:

$$c_{tf-idf}(W_t) = \sum_{w_1 \neq w_2} \log\left(\frac{\sum_d \text{tf-idf}(w_1, d)\text{tf-idf}(w_2, d)+\epsilon}{\sum_d \text{tf-idf}(w_1, d)}\right) \tag{3}$$

Donde W_t es el conjunto de palabras clave y el valor tf-idf se calcula según:

$$\text{tf-idf}(w, d) = \left(\frac{1}{2} + \frac{f(w,d)}{\max_{w'} f(w',d)}\right) \log\left(\frac{|D|}{|\{d' \in D : w \in d'\}|}\right) \tag{4}$$

Aquí, $f(w, d)$ es la frecuencia del término en el documento, normalizada por la frecuencia máxima. En tercer lugar, se consideró la **Diversidad (DIV)** [Dieng *et al.*, 2020], que calcula el porcentaje de palabras únicas dentro de los términos representativos, penalizando la redundancia. Finalmente, se midió la **Proximidad de Embeddings (WEP)** [Nikolenko, 2016], que evalúa la compactación semántica en el espacio vectorial mediante la similaridad coseno promedio:

$$\text{WEP}(W_t) = \frac{1}{|W_t|(|W_t|-1)} \sum_{w_1 \neq w_2} \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}||||\mathbf{B}||} \tag{5}$$

Donde **A** y **B** son los vectores de embeddings correspondientes a las palabras. Simultáneamente a estas métricas semánticas, se evaluó la estructura geométrica de los clústeres generados mediante tres indicadores adicionales, los cuales permitieron validar si las agrupaciones en el espacio vectorial eran compactas y distinguibles entre sí.

En primer lugar, se calculó el **Silhouette Score (SS)** [Rousseeuw, 1987], una métrica que evalúa qué tan similar es un objeto a su propio clúster (cohesión) en comparación con otros clústeres (separación). Esta métrica se define matemáticamente en la Ecuación 6:

$$\text{SS}(i) = \frac{b(i) - a(i)}{\text{máx}\{a(i), b(i)\}} \tag{6}$$

Donde $a(i)$ representa la distancia promedio entre el punto i y todos los demás puntos dentro del mismo clúster, indicando el grado de cohesión local. Por otro lado, $b(i)$ corresponde a la distancia promedio entre el punto i y todos los puntos del clúster vecino más cercano (el clúster distinto al de i que minimiza esta distancia), representando la separación externa. El valor resultante oscila entre -1 y 1, donde valores cercanos a 1 indican que el punto está bien emparejado con su propio clúster y mal emparejado con los vecinos.

En segundo lugar, se utilizó el **Índice Calinski-Harabasz (CH)** [Caliński y Harabasz, 1974], también conocido como Criterio de Ratio de Varianza. Este índice cuantifica la calidad del agrupamiento relacionando la dispersión entre los grupos con la dispersión dentro de los grupos, tal como se muestra en la Ecuación 7:

$$\text{CH} = \frac{\sum_i^k n_i \cdot ||\mu_i - \mu||^2}{\sum_i^k \sum_{x \in C_i} ||x - \mu_i||^2} \cdot \frac{N - k}{k - 1} \tag{7}$$

En esta formulación, el numerador representa la suma de cuadrados entre grupos (dispersión inter-clúster), donde n_i es el número de elementos en el clúster i, μ_i es el centroide del clúster i, y μ es el centroide global de todo el conjunto de datos. El denominador representa la suma de cuadrados dentro de los grupos (dispersión intra-clúster), calculada sumando las distancias cuadráticas de cada punto x respecto al centroide μ_i de su clúster C_i. Finalmente,

el término $\frac{N-k}{k-1}$ actúa como un factor de penalización y normalización, donde N es el número total de observaciones y k el número de clústeres formados. Un valor CH más alto indica clústeres mejor definidos (más densos y más separados).

Finalmente, se aplicó la métrica **Beta CV** [Kul *et al.*, 2018], la cual equilibra las magnitudes de cohesión y separación normalizando por el número de pares de puntos involucrados, lo que la hace robusta frente a variaciones en el tamaño de los clústeres. Su cálculo se describe en la Ecuación 8:

$$\beta = \frac{(\sum \text{dist}_{\text{in}}/n_{\text{in}})}{(\sum \text{dist}_{\text{out}}/n_{\text{out}})} \tag{8}$$

En esta ecuación, el numerador calcula la distancia intra-clúster promedio, donde $\sum \text{dist}_{\text{in}}$ es la suma de las distancias entre todos los pares de puntos que pertenecen al mismo clúster, y n_{in} es el conteo total de dichos pares intra-clúster. El denominador calcula la distancia inter-clúster promedio, donde $\sum \text{dist}_{\text{out}}$ es la suma de las distancias entre todos los pares de puntos que pertenecen a clústeres diferentes, y n_{out} es el conteo total de pares inter-clúster. A diferencia de las métricas anteriores, para Beta CV un valor más pequeño indica una mejor calidad de agrupamiento, ya que implica distancias internas pequeñas (alta cohesión) y distancias externas grandes (alta separación).

Una vez definido el marco de evaluación, se procedió a la fase experimental aprovechando la modularidad de BERTopic mediante una búsqueda de grilla (*grid search*) extensiva. En esta etapa se experimentó con dos modelos de lenguaje para la generación de embeddings: *Llama-2-13B-Chat*, representando a los modelos abiertos, y *text-embedding-3-large* de OpenAI, representando el estado del arte comercial. Asimismo, para la reducción de dimensionalidad con UMAP, se varió explícitamente el número de vecinos ($n\_neighbors \in \{3, 5, 10, 15, 30, 60, 100\}$) y el número de componentes ($n\_components \in \{2, 4, 6, 8, 10, 30, 100, 500, 1000\}$). Finalmente, se contrastaron dos algoritmos de agrupamiento explorando configuraciones específicas: para K-Means se varió el número de grupos ($k \in \{30, 35, 40, 45, 50, 55, 60, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85\}$), y para HDBSCAN se varió el tamaño mínimo de clúster ($min\_cluster \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 20, 25\}$), resultando en la evaluación de más de 4.000 variantes experimentales.

La selección de la mejor configuración no fue trivial dada la naturaleza conflictiva de las métricas. Para resolver esto, se implementó un sistema de ranking basado en el **Criterio de Borda Pesado con Penalización por Inestabilidad**. Si bien el criterio de Borda tradicional asigna puntos según la posición relativa de una variante, en este estudio se refinó incorporando la **desviación estándar** de las métricas de calidad de tópicos como factor de penalización. Como se puede ver en la Ecuación 9, la modificación se divide en dos aspectos: primero, la inclusión de la desviación estándar de cada una de las métricas semánticas (las cuales se ordenan en la tabla de posición de menor a mayor, donde una menor desviación estándar refleja una mejor variante); y segundo, en la distinta ponderación que se le asigna a cada una de las

métricas semánticas (8) y a las métricas de agrupamiento (3). En la ecuación "Borda$(...)$" refiere a la suma del puntaje obtenido al aplicar el criterio de Borda considerando las métricas dentro del paréntesis. Entonces, en resumen, para una variante el puntaje de Borda pesado con penalización por inestabilidad se calcula ponderando las 8 métricas semánticas (las 4 descritas junto a sus 4 desviaciones estándar que también son rankeadas) y las 3 métricas de agrupación, dividiendo la ponderación total equitativamente ($50\,\%$ 0 $0{,}5$ según se ve en la ecuación) entre familias de métricas. Esto permitió priorizar configuraciones que no solo tuvieran altos promedios de coherencia, sino que fueran *estables*, evitando variantes que produjeran una mezcla de tópicos excelentes con tópicos "basura".

$$w\text{Borda} = \frac{0{,}5}{8}\text{Borda}(\text{NPMI}, \Delta\text{NPMI}, \text{DIV}, \Delta\text{DIV}, \text{C-TF-IDF}, \Delta\text{C-TF-IDF}, \text{WEP}, \Delta\text{WEP}) + \frac{0{,}5}{3}\text{Borda}(\text{SS}, \text{CH}, \beta) \tag{9}$$

El análisis de los resultados, sintetizado en la Tabla 1, arrojó hallazgos determinantes. En primer lugar, se observó una superioridad consistente del algoritmo **HDBSCAN** sobre K-Means. HDBSCAN, al ser un algoritmo basado en densidad, tiene la capacidad de identificar y excluir "ruido" ("outliers") que no pertenece a ningún clúster denso; esta característica resultó crítica, pues al no forzar la clasificación de frases ambiguas, los clústeres resultantes mantuvieron una pureza semántica superior. En segundo lugar, respecto al modelo de embeddings, aunque *Llama-2* logró picos altos en métricas individuales, mostró una alta volatilidad. En contraste, el modelo **text-embedding-3-large** de OpenAI demostró una estabilidad superior y dominó el ranking final al aplicar la penalización por desviación estándar.

Tabla 1: Las mejores variantes para la representación por frases según el criterio de Borda Pesado con penalización por desviación estándar.

Modelo	N_VEC	N_COM	TAM	NPMI ↑	DIV ↑	C-TF-IDF ↑	WEP ↓	SS ↑	CH ↑	$\beta\downarrow$	w Borda ↑
OpenAI	15	4	8	0.243	65.8 %	79.5	0.2724	0.0806	13.15	0.812	**4251.08**
OpenAI	30	4	7	0.253	65.3 %	79.3	0.2730	0.0891	12.89	0.809	4240.04
OpenAI	30	2	8	0.235	66.7 %	75.0	0.2705	0.0744	13.21	0.808	4237.45
OpenAI	15	4	7	0.255	67.9 %	82.9	0.2714	0.0867	12.39	0.808	4221.50

Finalmente, la interpretación cualitativa de los tópicos generados por la variante ganadora (Tabla 2) confirmó la validez de las métricas seleccionadas. Los tópicos con mayor puntaje revelaron conceptos técnicos claros como "Deuda Técnica" y "Pair Programming", mientras que incluso los tópicos con menor puntaje mantuvieron coherencia temática (e.g., "Reuniones diarias"), validando así la decisión de priorizar la estabilidad del modelo mediante el uso de embeddings comerciales avanzados y filtrado de ruido por densidad.

El detalle de cada uno de los tópicos y sus palabras clave se puede encontrar en el Anexo en la Tabla 11.

Tabla 2: Ejemplo de Tópicos (Mejores y Peores) generados por la mejor variante (OpenAI + HDBSCAN).

Ranking	Etiqueta Inferida	Palabras Clave (c-TF-IDF)
Top 1	Deuda Técnica	medidas, documentando, abordar, **deuda**, **técnica**
Top 2	Pair Programming	pantalla, programación, sesiones, **pair**, **programming**
Top 3	Criterios Aceptación	descritos, algunos, **criterio**, criterios, aceptación
...	...	...
Bottom 2	Clima Laboral	algo, buena, ambiente, mantener, onda
Bottom 1	Daily Meetings	meetings, mantener, **daily**, diarias, reuniones

4.3.2. Clasificación de las frases de las Retrospectivas de Sprint con LLMs

De manera complementaria al análisis inductivo realizado previamente, y siguiendo el flujo metodológico deductivo presentado en la Figura 2, se implementó un enfoque de clasificación supervisada para estructurar el conocimiento contenido en las retrospectivas. A diferencia del modelado de tópicos donde los temas emergen de los datos, este enfoque buscó clasificar cada frase dentro de un marco teórico predefinido de 23 etiquetas temáticas, modelando el análisis deductivo como una tarea de clasificación multi-etiqueta asistida por Grandes Modelos de Lenguaje (LLMs). El objetivo fue evaluar la capacidad de estos modelos para simular el juicio de un experto humano en la identificación de temas de Ingeniería de Software.

El proceso metodológico inició con la definición formal de las etiquetas y la construcción de un conjunto de datos de referencia (Gold Standard) por parte de anotadores expertos. Aunque las etiquetas iniciales provinieron de la literatura general, su consolidación se basó fundamentalmente en el marco propuesto en [Palopak y Huang, 2024], quienes vinculan la adherencia a los principios ágiles con distintas dimensiones de éxito en el desarrollo de software. Este enfoque resultó crucial para el contexto del curso capstone, ya que permitió adaptar dichas dimensiones de éxito profesional —tales como la calidad del producto, la satisfacción de los interesados y la eficiencia del equipo— a la realidad educativa. Así, tras un refinamiento iterativo con los instructores del curso para capturar la naturaleza multifacética de las discusiones estudiantiles, se estableció una taxonomía de 23 etiquetas temáticas que abarcan desde aspectos técnicos rigurosos hasta las dinámicas humanas críticas identificadas en el estudio base. Para asegurar la consistencia semántica y mitigar la ambigüedad inherente al lenguaje natural, se generó un diccionario de definiciones precisas (detallado en la Tabla 3) que actuó como estándar tanto para los anotadores humanos como para el modelo de lenguaje.

Tabla 3: Definición de las 23 etiquetas temáticas utilizadas para la clasificación.

Etiqueta	Descripción
Motivación	La frase se refiere explícitamente a la actitud, entusiasmo, responsabilidad o motivación del equipo para trabajar y si estos factores son buenos, malos o necesitan mejorarse.
Carga académica	La frase se relaciona con cómo la carga académica u otras responsabilidades académicas que tienen los miembros del equipo afectan su trabajo o son manejadas por el equipo.
Criterios de aceptación	La frase se relaciona con cómo se realiza la revisión del cumplimiento de los criterios de aceptación o la definición de "hecho" (*done*) de las tareas del proyecto (o si no se hace). Conocer los criterios de aceptación implica que el equipo tiene una definición transversal de cómo debe comportarse el sistema y busca comprobar que este comportamiento se logre.
Reporte de estado	La frase se relaciona con la forma que adoptan los miembros del equipo para reportar su estado al resto del equipo de desarrollo, ya sea relacionado con sus tareas o los problemas que tienen, si este reporte se hizo o no, cómo se hizo (usando o no herramientas), si fue bueno o malo o si necesita mejorarse.
Apoyo	Como los estudiantes son mayoritariamente novatos en desarrollo de software, este tema se relaciona con cómo se dio o recibió el apoyo o ayuda específicamente entre miembros del equipo al trabajar, principalmente relacionado con tener problemas con las tareas del proyecto.
Comunicación	La frase se relaciona con cómo se realiza la comunicación entre los miembros del equipo, si es buena, mala o necesita mejorarse.
Cara a cara	La frase se relaciona con cómo (si es que ocurre) se llevan a cabo las reuniones presenciales, específicamente si se usan para trabajar o discutir aspectos del proyecto, teniendo comunicación directa cara a cara.
Reuniones	La frase se relaciona con cómo se llevan a cabo las reuniones que el equipo organizó para discutir aspectos del proyecto, si son buenas, malas o necesitan mejorarse.
Relaciones	La frase se relaciona con describir la relación, confianza o afinidad entre los miembros del equipo. Este tema se enfoca en cómo son las interacciones entre los miembros fuera de la responsabilidad del proyecto, incluyendo pero no limitado a tener buen trato, declarar confianza mutua, si disfrutan trabajar juntos, etc.
Docentes	La frase se relaciona con las interacciones o cómo se llevan a cabo las interacciones con los instructores o ayudantes del curso que están a cargo del proyecto y evalúan el trabajo de los estudiantes.

Etiqueta	Descripción
Clientes	La frase se relaciona con las interacciones o cómo se llevan a cabo las interacciones con el cliente u otros interesados fuera de los instructores del curso (pueden ser otros profesores).
Diseño	La frase se relaciona con el proceso de diseño del sistema que el equipo siguió, sigue o seguirá para organizar los componentes de software del sistema o su diseño UI/UX, incluyendo el uso de cualquier tipo de herramienta para ayudar al equipo con el proceso de diseño.
Ritmo	La frase se relaciona con el ritmo que tuvo o tiene el equipo, la consistencia de la completitud del trabajo por parte de los miembros (si es buena o mala) o el progreso de las tareas del proyecto (si es bueno o malo) y si tiene obstáculos o no.
Asignación de tareas	La frase se relaciona con cómo se realiza la definición, asignación o distribución de tareas a realizar en el proyecto, si es buena, mala o necesita mejorarse, cómo se realiza la comunicación de la asignación, si hay uso de herramientas para ayudar al equipo específicamente con esto, y cómo el equipo se organiza para cubrir las responsabilidades.
Priorización de tareas	La frase se relaciona con cómo se realiza la priorización de las tareas del proyecto por parte de los miembros del equipo, si es buena, mala o necesita mejorarse.
Alcance	La frase se relaciona con la estimación del alcance o el trabajo planificado (o Sprint) incluyendo el enfoque o la intención detrás de las historias de usuario o criterios de aceptación y el cumplimiento de lo planificado.
Frecuencia de integración	La frase se relaciona con qué tan a menudo el equipo integra el código o el trabajo realizado por cada uno de los miembros en una versión unificada, si es buena, mala o necesita mejorarse.
Experiencia técnica	La frase se relaciona con la experiencia del equipo en el proyecto o la búsqueda de conocimiento relacionado con el proyecto y las tecnologías involucradas.
Tecnologías	La frase se relaciona con cómo está usando el equipo las tecnologías o herramientas elegidas para desarrollar el proyecto, si es bueno, malo o necesita mejorarse. Se refiere a declaraciones sobre tecnologías o herramientas específicas y sus marcas comerciales.
Calidad de software	La frase se relaciona con la calidad del código del sistema que los estudiantes implementaron. Incluye aspectos de la interfaz que son buenos o malos, la calidad general del sistema o de las funcionalidades, y el *testing* específico de la calidad del software.

Etiqueta	Descripción
Implementación	La frase se relaciona con cómo los miembros se coordinan para escribir el código o construir el sistema, considerando herramientas, técnicas, estándares o métodos. Incluye la comunicación de decisiones sobre la implementación o el comportamiento del sistema para continuar la construcción.
Integración	La frase se relaciona con cómo los miembros se coordinan para integrar su trabajo en el sistema de versiones del equipo, considerando el uso de herramientas, técnicas, estándares u otros métodos relacionados con formas de integrar el código a un sistema de versiones.
Propiedad del código	La frase se relaciona con qué tan conocedores son los miembros del equipo sobre todos los módulos, componentes, partes o secciones del sistema y su comportamiento.

Posteriormente, como se señaló en el párrafo anterior, se llevó a cabo un proceso de etiquetado experto independiente y consensuado, en donde cada experto a través de la plataforma "Label Studio"[1] asignó a un mismo conjunto de 285 frases los Temas a los que se refiere cada una. Posterior a que cada experto terminara con todas las frases de este conjunto, se discutió el etiquetado asignado a las frases en donde habían discrepancias, explicando las razones que llevaron a cada uno a asignar los Temas que asignaron. Luego de haber escuchado las razones de ambos, los anotadores acordaron un etiquetado final para la frase, conteniendo entonces los Temas que ambos consensuaron para estar presentes en cada una de estas frases.

Se dividió el corpus total de frases en tres subconjuntos estratégicos: Entrenamiento, Validación y Prueba. Esta separación de las frases en tres conjuntos se motiva en adaptar al caso de los LLM la aproximación tradicional para la evaluación y entrenamiento de los modelos de aprendizaje estadístico en el área de aprendizaje automático: las frases y sus etiquetas que estén contenidas dentro del conjunto de entrenamiento puede ser usadas como ejemplos por el modelo para imitar de mejor manera la respuesta deseada mediante el aprendizaje en contexto (ICL), todo esto dentro del prompt que se le entrega para hacer las predicciones de las etiquetas; el conjunto de validación y de pruebas, por otro lado son usados para la evaluación del rendimiento del modelo en frases en que no se les ha presentado la etiqueta. La diferencia entre los conjuntos de validación y pruebas radica en el propósito de la evaluación: el rendimiento del modelo en el primero sirve para que los investigadores puedan hacer ajustes iterativos en dirección de mejorar el rendimiento del modelo, finalizando, en este caso, en la definición de las instrucciones que maximizan su desempeño en este conjunto; el conjunto de pruebas, en cambio, como no se ha visto ni evaluado el rendimiento del modelo sobre esas frases, es utilizado como un reporte del rendimiento futuro del modelo por sobre la predicción de las etiquetas de las frases que no ha visto y en las que no fue precisamente ajustado para rendir de la mejor manera posible, lo cual es el caso de esta tesis

[1] https://labelstud.io/

al buscar la definición de un método de predicción inteligente que permita su aplicación en frases de retrospectivas dentro de la repetición del curso estudiado.

Explicado esto, el análisis de la distribución de las 23 etiquetas, presentado en la Tabla 4, reveló el desbalance natural del dominio. Temas operativos como “Reuniones” y “Tecnologías” predominaron consistentemente, mientras que etiquetas de mayor complejidad abstracta, como “Priorización de tareas” o “Frecuencia de integración”, tuvieron una representación marginal.

Tabla 4: Distribución de etiquetas en los conjuntos de Entrenamiento, Validación y Prueba.

Etiqueta	**Entrenamiento ($N = 62$)**	**Validación ($N = 109$)**	**Prueba ($N = 114$)**
Motivación	6	10	11
Carga académica	0	3	4
Criterios de aceptación	3	11	11
Reporte de estado	8	14	15
Soporte	1	3	6
Comunicación	3	7	8
Cara a cara	2	2	2
Reuniones	7	18	16
Relaciones	5	9	7
Profesores	1	2	1
Clientes	2	4	4
Diseño	4	4	6
Ritmo	7	14	11
Asignación de tareas	3	10	9
Priorización de tareas	2	1	1
Alcance	4	9	9
Frecuencia de integración	1	3	2
Experiencia	2	7	8
Tecnologías	8	15	16
Calidad de software	6	9	10
Implementación	2	4	4
Integración	9	10	16
Propiedad del código	0	0	1

De todos modos, la distribución desbalanceada que se observa en la Tabla 4 también es consecuencia de la cantidad de frases que pudieron ser etiquetadas en el tiempo (285). Esta restricción, sumada a la frecuencia natural de los temas en las discusiones estudiantiles, provocó la escasez de soporte de la etiqueta *Propiedad del código*, la cual no registró ocurrencia alguna en el conjunto de validación (ni en el de entrenamiento). Esta ausencia total de instancias positivas imposibilitó la aplicación de la metodología de Ingeniería de Prompts basada en el rendimiento empírico, dado que no existía una base estadística en la etapa de validación que permitiera discernir qué configuración de atributos resultaba superior para la detección de este tema específico. Es por esto que dicha etiqueta fue excluida del análisis.

Con esto y una vez establecido el conjunto de datos, se procedió a la fase de Ingeniería de Prompts para optimizar el rendimiento del modelo gratuito “GPT-OSS-20B”, el fue se-

leccionado para esta etapa por la facilidad que ofrece en su instalación local y uso a través de la aplicación "LMStudio" [2], junto al buen rendimiento que ha mostrado en la literatura [Bi *et al.*, 2025]. Por otro lado, y dado que los LLMs son sensibles a la estructura de las instrucciones, para esta etapa se diseñó un experimento factorial completo (2^4) para evaluar el impacto de cuatro componentes estratégicos en el *prompt*, resultando en 16 configuraciones experimentales distintas. El objetivo de esto es definir que estrategias maximizan el rendimiento del modelo dentro de la tarea, y las estrategias evaluadas fueron:

1. **Uso de Ejemplos (Few-Shot):** Se evaluó la inclusión de tres ejemplos seleccionados del conjunto de entrenamiento para guiar al modelo mediante aprendizaje en contexto (*In-Context Learning*).
2. **Razonamiento (Reasoning):** Se instruyó al modelo para generar una cadena de pensamiento (*Chain-of-Thought*) antes de emitir su veredicto final, subdividiendo la frase y analizando semánticamente sus partes.
3. **Exclusiones:** Esta estrategia consistió en incluir explícitamente en la descripción de la tarea las definiciones de *otros* temas distintos al evaluado. El propósito fue indicar al modelo que, si la frase se alineaba mejor con alguna de esas definiciones alternativas, debía excluirla de la clasificación actual.
4. **Descripciones Generadas (Descriptions):** Se incorporó una descripción enriquecida de cada etiqueta generada previamente por el mismo modelo (auto-ayuda). Para esto, el modelo analizó los datos de entrenamiento y sus etiquetas expertas, sintetizando una descripción que capturaba los patrones que él mismo identificaba en los datos.

Basado en la presencia o ausencia de estos componentes, la composición específica de cada prompt evaluado es la siguiente:

P1: Zero-Shot + Exclusiones. (Sin ejemplos, sin razonamiento, con exclusiones, descripción base).

P2: Zero-Shot + Exclusiones + Descripciones. (Sin ejemplos, sin razonamiento, con exclusiones, descripción generada).

P3: Zero-Shot Base. (Sin ejemplos, sin razonamiento, sin exclusiones, descripción base).

P4: Zero-Shot + Descripciones. (Sin ejemplos, sin razonamiento, sin exclusiones, descripción generada).

P5: Zero-Shot + Razonamiento + Exclusiones. (Sin ejemplos, con CoT, con exclusiones, descripción base).

[2] https://lmstudio.ai/

P6: Zero-Shot + Razonamiento + Exclusiones + Descripciones. (Sin ejemplos, con CoT, con exclusiones, descripción generada).

P7: Zero-Shot + Razonamiento. (Sin ejemplos, con CoT, sin exclusiones, descripción base).

P8: Zero-Shot + Razonamiento + Descripciones. (Sin ejemplos, con CoT, sin exclusiones, descripción generada).

P9: Few-Shot + Exclusiones. (Con ejemplos, sin razonamiento, con exclusiones, descripción base).

P10: Few-Shot + Exclusiones + Descripciones. (Con ejemplos, sin razonamiento, con exclusiones, descripción generada).

P11: Few-Shot Base. (Con ejemplos, sin razonamiento, sin exclusiones, descripción base).

P12: Few-Shot + Descripciones. (Con ejemplos, sin razonamiento, sin exclusiones, descripción generada).

P13: Few-Shot + Razonamiento + Exclusiones. (Con ejemplos, con CoT, con exclusiones, descripción base).

P14: Few-Shot + Razonamiento + Exclusiones + Descripciones. (Con ejemplos, con CoT, con exclusiones, descripción generada).

P15: Few-Shot + Razonamiento. (Con ejemplos, con CoT, sin exclusiones, descripción base).

P16: Few-Shot + Razonamiento + Descripciones. (Con ejemplos, con CoT, sin exclusiones, descripción generada).

La evaluación del rendimiento se realizó sobre el conjunto de validación utilizando métricas robustas frente al desbalance de clases observado. Se calculó el **Micro F1 Score**, que agrega las contribuciones de todas las clases tratando cada instancia por igual, y el **Macro F1 Score**, que calcula el promedio simple de los F1 Scores por clase, otorgando igual peso a las etiquetas minoritarias. El Micro F1 se define en la Ecuación 10:

$$\text{Micro-F1} = 2 \cdot \frac{\text{Micro-Precision} \cdot \text{Micro-Recall}}{\text{Micro-Precision} + \text{Micro-Recall}} \tag{10}$$

Donde la Micro-Precisión (Ecuación 11) y el Micro-Recall (Ecuación 12) se calculan sobre la suma total de verdaderos positivos (TP), falsos positivos (FP) y falsos negativos (FN) a través de las n clases:

$$\text{Micro-Precision} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FP_i)} \tag{11}$$

$$\text{Micro-Recall} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} (TP_i + FN_i)} \tag{12}$$

Para consolidar los resultados y establecer un ordenamiento definitivo que balancee las fortalezas de cada configuración, se introdujo el criterio de Borda (ver Tabla 5). Este puntaje se calcula sumando, para cada configuración, la cantidad de otras combinaciones a las que logra superar estrictamente tanto en *Micro F1* como en *Macro F1*. De esta manera, el índice Borda recompensa la consistencia, penalizando aquellos modelos que pueden tener un alto desempeño en una métrica a costa de sacrificar la otra. Bajo este enfoque integral, la configuración **P8** (Ranking 1) se confirma como la más robusta, empatando en puntaje con la configuración completa **P14** (Ranking 2) pero priorizándose por su mayor eficiencia y simplicidad.

Un análisis de los resultados permite extraer hallazgos reveladores sobre la naturaleza del aprendizaje en contexto del modelo. La superioridad de la configuración ganadora (**P8**: *Zero-Shot + Razonamiento + Descripciones*) por sobre variantes más complejas sugiere que la calidad de la instrucción semántica prevalece sobre la cantidad de evidencia mostrada. Al dotar al modelo de una definición precisa del concepto (generada por él mismo para maximizar su comprensión) y forzarlo a explicitar su proceso deductivo, se logra una alineación cognitiva más robusta que la que ofrecen los ejemplos estáticos. De hecho, se observa un fenómeno notable al comparar la combinación **P8** (Ranking 1) con la **P16** (Ranking 14): ambas configuraciones son idénticas salvo por la inclusión de ejemplos en la segunda. Sorprendentemente, la adición de ejemplos provocó una caída drástica del Macro F1 del 81.66 % al 74.75 %. Esto podría revelar que el conjunto de ejemplos que se le entregan al modelo, si no cubren perfectamente la varianza del problema, pueden sesgarlo hacia patrones específicos y superficiales presentes en ellos, distrayéndolo de la definición general y abstracta que provee la descripción.

Por otro lado, el análisis de las configuraciones con peor desempeño, como la **P1** (*Solo Exclusiones*, Ranking 15), ilumina la ineficacia de las restricciones negativas cuando no existe un anclaje positivo fuerte. Decirle al modelo "qué no es" una etiqueta (Exclusiones) sin ofrecerle una definición clara de "cómo es" (Descripciones) ni un mecanismo para procesarlo (Razonamiento), resulta en una peor inferencia para este caso. Asimismo, se evidencia que el Razonamiento por sí solo no es una bala de plata; necesitando de información de calidad para operar.

Si bien el análisis del rendimiento promedio presentado anteriormente permite identificar tendencias generales sobre la efectividad de las configuraciones de *prompting*, es indispensable examinar el comportamiento a nivel de etiqueta individual. La naturaleza semántica de cada categoría —desde conceptos técnicos tangibles hasta dinámicas interpersonales sutiles— provoca que la respuesta del modelo ante elementos como el razonamiento o el uso de descripciones autogeneradas varíe significativamente. Las Tablas 6 y 7 presentan un desglose exhaustivo del *F1-Score* obtenido por cada etiqueta bajo las 16 configuraciones evaluadas.

Tabla 5: Rendimiento en el conjunto de validación ordenado según el criterio de Borda. El puntaje Borda representa la suma de combinaciones superadas en Micro y Macro F1, ofreciendo una métrica unificada de rankeo.

Ranking	ID	Ejemplos	Razonamiento	Exclusiones	Descripciones	Micro F1 (%)	Macro F1 (%)	Borda
1	**P8**	No	Sí	No	Sí	80.36	81.66	29
2	**P14**	Sí	Sí	Sí	Sí	81.74	80.88	29
3	**P15**	Sí	Sí	No	No	79.33	79.80	25
4	**P6**	No	Sí	Sí	Sí	79.88	78.82	24
5	**P4**	No	No	No	Sí	78.92	79.26	22
6	**P12**	Sí	No	No	Sí	78.96	77.06	18
7	**P2**	No	No	Sí	Sí	77.58	78.76	16
8	**P13**	Sí	Sí	Sí	No	77.78	77.65	15
9	**P5**	No	Sí	Sí	No	78.08	76.87	15
10	**P3**	No	No	No	No	77.10	77.94	13
11	**P9**	Sí	No	Sí	No	77.09	76.66	8
12	**P11**	Sí	No	No	No	75.75	76.24	5
13	**P10**	Sí	No	Sí	Sí	77.10	75.69	7
14	**P16**	Sí	Sí	No	Sí	77.91	74.75	10
15	**P1**	No	No	Sí	No	75.96	74.54	2
16	**P7**	No	Sí	No	No	75.45	74.72	1

Al analizar los resultados detallados, se observan patrones distintivos en función de la complejidad de la etiqueta. Categorías robustas y bien delimitadas, como *Profesores* o *Carga académica*, mostraron una notable indiferencia ante la variación del *prompt*, alcanzando rendimientos óptimos (incluso del 100 %) con configuraciones mínimas (**P3**). En contraste, etiquetas que involucran dinámicas humanas complejas o definiciones técnicas amplias, como *Comunicación*, *Ritmo* y *Tecnologías*, exigieron la activación de todos los componentes del prompt (Ejemplos, Razonamiento, Exclusiones y Descripciones) para maximizar su detección (**P14** y **P16**). El caso de *Comunicación* es particularmente crítico: mientras que la configuración completa alcanza un 73.68 % de F1, la ausencia de descripciones o razonamiento puede desplomar el rendimiento hasta un 20 %, evidenciando la alta dependencia contextual de esta categoría.

Por otro lado, etiquetas como *Diseño*, *Implementación* y *Relaciones* se beneficiaron específicamente de la combinación de Razonamiento y Descripciones (**P8**), sugiriendo que el modelo requiere "pensar" y tener una definición clara para distinguir estos temas, pero no necesariamente requiere ejemplos *few-shot*. Finalmente, destaca el caso de *Alcance*, donde el uso de exclusiones explícitas (**P2**) resultó fundamental para evitar falsos positivos, alcanzando su máximo rendimiento (73.68 %) y superando ampliamente a configuraciones que, aunque más complejas, introducían ruido (como el **P12**, que cae al 18.18 %).

Con base en el análisis exhaustivo del rendimiento por etiqueta presentado anteriormente, se determinó que utilizar una única configuración de *prompt* para todo el conjunto de datos sería subóptimo. La variabilidad en la naturaleza semántica de las etiquetas exige estrategias

Tabla 6: Desempeño detallado (F1 Score) por etiqueta para los Prompts 1 al 8. En **negrita** el máximo rendimiento global obtenido para la etiqueta.

Etiqueta	P1 (No Ex, No Rea Excl, No Desc)	P2 (No Ex, No Rea Excl, Desc)	P3 (No Ex, No Rea No Excl, No Desc)	P4 (No Ex, No Rea No Excl, Desc)	P5 (No Ex, Rea Excl, No Desc)	P6 (No Ex, Rea Excl, Desc)	P7 (No Ex, Rea No Excl, No Desc)	P8 (No Ex, Rea No Excl, Desc)
Motivación	76.92 %	90.91 %	80.00 %	**95.24 %**	83.33 %	90.91 %	80.00 %	90.00 %
Carga académica	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**
Crit. de aceptación	80.00 %	66.67 %	80.00 %	58.82 %	84.21 %	80.00 %	80.00 %	58.82 %
Reporte de estado	76.92 %	78.57 %	92.31 %	80.00 %	88.00 %	88.89 %	88.89 %	88.00 %
Soporte	60.00 %	**66.67 %**	60.00 %	**66.67 %**	60.00 %	60.00 %	60.00 %	**66.67 %**
Comunicación	66.67 %	63.16 %	63.16 %	63.16 %	62.50 %	20.00 %	63.16 %	60.00 %
Cara a cara	50.00 %	**66.67 %**	**66.67 %**	50.00 %	**66.67 %**	**66.67 %**	**66.67 %**	**66.67 %**
Reuniones	91.89 %	97.14 %	94.74 %	**100.00 %**	97.30 %	**100.00 %**	97.30 %	**100.00 %**
Relaciones	**94.12 %**	88.89 %	87.50 %	**94.12 %**	**94.12 %**	**94.12 %**	87.50 %	**94.12 %**
Profesores	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**
Clientes	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	88.89 %	**100.00 %**	88.89 %	**100.00 %**
Diseño	80.00 %	80.00 %	60.00 %	88.89 %	75.00 %	72.73 %	60.00 %	**100.00 %**
Ritmo	68.97 %	58.33 %	68.75 %	70.97 %	70.97 %	76.92 %	66.67 %	70.97 %
Asignación de tareas	88.89 %	82.35 %	88.89 %	94.74 %	82.35 %	82.35 %	82.35 %	94.74 %
Priorización de tareas	66.67 %	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**
Alcance	66.67 %	**73.68 %**	47.62 %	55.56 %	63.64 %	70.00 %	58.82 %	66.67 %
Frecuencia de integración	80.00 %	80.00 %	**100.00 %**	80.00 %	80.00 %	80.00 %	80.00 %	80.00 %
Experiencia	72.73 %	66.67 %	**83.33 %**	72.73 %	60.00 %	72.73 %	72.73 %	72.73 %
Tecnologías	71.43 %	74.07 %	78.57 %	74.07 %	78.57 %	74.07 %	69.23 %	74.07 %
Calidad de software	55.56 %	63.16 %	63.16 %	63.16 %	63.16 %	**66.67 %**	58.82 %	**66.67 %**
Implementación	40.00 %	75.00 %	57.14 %	75.00 %	40.00 %	**85.71 %**	40.00 %	**85.71 %**
Integración	66.67 %	75.00 %	57.14 %	75.00 %	66.67 %	66.67 %	57.14 %	75.00 %

Tabla 7: Continuación: Desempeño detallado (F1 Score) por etiqueta para los Prompts 9 al 16. En **negrita** el máximo rendimiento global obtenido para la etiqueta.

Etiqueta	P9 (Ex, No Rea Excl, No Desc)	P10 (Ex, No Rea Excl, Desc)	P11 (Ex, No Rea No Excl, No Desc)	P12 (Ex, No Rea No Excl, Desc)	P13 (Ex, Rea Excl, No Desc)	P14 (Ex, Rea Excl, Desc)	P15 (Ex, Rea No Excl, No Desc)	P16 (Ex, Rea No Excl, Desc)
Motivación	80.00 %	83.33 %	76.92 %	78.26 %	74.07 %	83.33 %	83.33 %	80.00 %
Carga académica	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**	**85.71 %**
Crit. de aceptación	90.00 %	85.71 %	85.71 %	85.71 %	**95.24 %**	84.21 %	**95.24 %**	76.19 %
Reporte de estado	83.87 %	84.62 %	82.76 %	92.31 %	75.86 %	84.62 %	**92.86 %**	78.57 %
Soporte	60.00 %	60.00 %	60.00 %	**66.67 %**	60.00 %	60.00 %	60.00 %	**66.67 %**
Comunicación	60.00 %	66.67 %	57.14 %	66.67 %	66.67 %	**73.68 %**	60.00 %	20.00 %
Cara a cara	**66.67 %**	**66.67 %**	**66.67 %**	**66.67 %**	**66.67 %**	**66.67 %**	**66.67 %**	50.00 %
Reuniones	**100.00 %**	97.30 %	97.30 %	**100.00 %**	**100.00 %**	**100.00 %**	97.30 %	**100.00 %**
Relaciones	80.00 %	80.00 %	76.19 %	84.21 %	84.21 %	84.21 %	**94.12 %**	84.21 %
Profesores	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**
Clientes	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**	**100.00 %**
Diseño	72.73 %	72.73 %	66.67 %	66.67 %	72.73 %	72.73 %	66.67 %	80.00 %
Ritmo	66.67 %	23.53 %	59.46 %	76.47 %	68.75 %	**83.87 %**	62.86 %	78.79 %
Asignación de tareas	90.00 %	90.00 %	84.21 %	94.74 %	94.74 %	88.89 %	**100.00 %**	**100.00 %**
Priorización de tareas	66.67 %	66.67 %	66.67 %	66.67 %	66.67 %	**100.00 %**	66.67 %	40.00 %
Alcance	63.64 %	70.00 %	72.73 %	18.18 %	69.57 %	70.00 %	66.67 %	72.73 %
Frecuencia de integración	80.00 %	80.00 %	80.00 %	80.00 %	80.00 %	80.00 %	**100.00 %**	80.00 %
Experiencia	76.92 %	**83.33 %**	**83.33 %**	**83.33 %**	**83.33 %**	**83.33 %**	**83.33 %**	**83.33 %**
Tecnologías	73.33 %	78.57 %	74.07 %	78.57 %	64.00 %	**82.76 %**	69.23 %	74.07 %
Calidad de software	60.87 %	63.64 %	60.00 %	54.55 %	**66.67 %**	63.16 %	52.63 %	52.63 %
Implementación	54.55 %	44.44 %	75.00 %	75.00 %	66.67 %	57.14 %	**85.71 %**	66.67 %
Integración	75.00 %	**82.35 %**	66.67 %	75.00 %	66.67 %	75.00 %	66.67 %	75.00 %

diferenciadas: mientras que categorías concretas se detectan fácilmente con instrucciones simples, conceptos abstractos requieren mayor contexto y restricciones.

Por consiguiente, para la fase de predicción final sobre la totalidad de las retrospectivas no etiquetadas y previamente la evaluación en el conjunto de pruebas, se asignó a cada etiqueta

la configuración de prompt específica que maximizó su F1-Score en el conjunto de validación. La distribución de configuraciones seleccionadas se detalla a continuación:

- **Configuración Base (P3):** Para etiquetas robustas y semánticamente inambiguas, la configuración mínima (sin ejemplos ni razonamiento) demostró ser suficiente y eficiente.
    - *Profesores* (100.0 %), *Carga académica* (85.71 %), *Cara a cara* (66.67 %).
- **Solo Descripciones (P4):** La adición de definiciones claras y auto generadas fue determinante para etiquetas que, aunque concretas, requieren desambiguación leve.
    - *Reuniones* (100.0 %), *Clientes* (100.0 %), *Priorización de tareas* (100.0 %), *Motivación* (95.24 %), *Soporte* (66.67 %).
- **Razonamiento y Descripciones (P8):** Etiquetas que implican procesos cognitivos o evaluación cualitativa se beneficiaron significativamente de la cadena de pensamiento (*Chain-of-Thought*) sin necesidad de ejemplos.
    - *Diseño* (100.0 %), *Relaciones* (94.12 %), *Implementación* (85.71 %), *Calidad de software* (66.67 %).
- **Configuración Completa (P14):** Las etiquetas más complejas y propensas a falsos positivos requirieron la activación de todos los componentes (Ejemplos, Razonamiento, Exclusiones y Descripciones).
    - *Criterios de aceptación* (95.24 %), *Ritmo* (83.87 %), *Tecnologías* (82.76 %), *Comunicación* (73.68 %).
- **Configuraciones Híbridas Específicas:** Se seleccionaron combinaciones particulares para etiquetas con comportamientos únicos:
    - **P16 (Ejemplos + Razonamiento + Descripciones):** Para *Asignación de tareas* (100.0 %) y *Frecuencia de integración* (100.0 %).
    - **P15 (Ejemplos + Razonamiento):** Específico para *Reporte de estado* (92.86 %).
    - **P10 (Ejemplos + Exclusiones + Descripciones):** Para *Integración* (82.35 %).
    - **P11 (Solo Ejemplos):** Para *Experiencia* (83.33 %), donde los ejemplos resultaron más efectivos que las definiciones teóricas.
    - **P2 (Exclusiones + Descripciones):** Crítico para *Alcance* (73.68 %), donde limitar el ámbito mediante exclusiones fue clave.

Esta selección granular asegura que la capacidad del modelo se canalice eficientemente, aplicando razonamiento complejo y ejemplos *few-shot* solo donde la ambigüedad de la etiqueta lo justifica, optimizando así la precisión global del sistema de clasificación.

Siguiendo la lógica experimental estándar en el aprendizaje automático, la selección de la mejor configuración de *prompt* para cada etiqueta no constituye el paso final. Dado que el objetivo último es aplicar estos modelos sobre un corpus de frases no etiquetadas, es imperativo validar el rendimiento de las configuraciones ganadoras en el conjunto de pruebas, el cual es independiente al proceso de experimentación y representativo del escenario futuro que enfrentará el modelo.

La Tabla 8 presenta el desempeño de cada etiqueta utilizando su configuración de prompt óptima aplicada sobre este conjunto de pruebas. Los resultados muestran una variabilidad esperada: mientras etiquetas como *Clientes*, *Profesores* y *Cara a cara* alcanzaron un rendimiento perfecto (F1 = 1.00), otras categorías más complejas como *Implementación* o *Priorización de tareas* mostraron dificultades significativas para generalizar las reglas inferidas. A nivel global, el modelo alcanzó un Micro-F1 de 0.7209 y un Macro-F1 de 0.7054, cifras que sugieren una capacidad moderada-alta para capturar la semántica de las etiquetas de interés dentro de las retrospectivas estudiantiles.

Al analizar en detalle la columna de *Soporte* en la Tabla 8, se hace evidente una limitación estadística crítica: varias etiquetas, como *Priorización de tareas* (n=1), *Frecuencia de integración* (n=2) o *Cara a cara* (n=2), cuentan con una cantidad insuficiente de ejemplos en el conjunto de pruebas para garantizar que las métricas de F1 sean un estimador confiable del comportamiento del modelo a gran escala. Un rendimiento perfecto o nulo en estos casos podría ser anecdótico y no reflejar la verdadera capacidad de generalización del LLM.

Para mitigar este riesgo y asegurar que las conclusiones extraídas del análisis masivo sean robustas, se implementó un mecanismo de verificación *post-hoc* basado en el juicio experto. Este procedimiento consistió en extraer una muestra aleatoria de frases etiquetadas por el modelo en el conjunto de datos final (diferente al conjunto de 285 frases ya etiquetadas) y someterlas a una revisión ciega (sin ver los Temas etiquetados por el modelo) por parte de un experto, el cual posteriormente anotaría los Temas que considera deberían ser asignados a cada una de estas frases. Respecto como fue seleccionado este conjunto nuevo de frases, se optó que, por cada etiqueta o Tema, se seleccione un conjunto aleatorio de 10 frases en donde el modelo marcó como presente el Tema y 10 más en donde el modelo marcó como ausente el Tema, así buscando contrastar con la anotación que hizo el experto a cada una de estas frases como el modelo reconoce la ausencia o presencia de cada Tema en un conjunto distinto al previamente etiquetado.

Durante la revisión, en donde se ocultó al experto la etiqueta predicha por el modelo para evitar sesgos de confirmación, el experto asignó las etiquetas reales (*Ground Truth*) a estas muestras, las cuales fueron posteriormente contrastadas con las predicciones generadas por el LLM. Los resultados de rendimiento (Precisión, Exhaustividad y F1-Score) para esta validación de sanidad se detallan en la Tabla 9.

Como se puede observar, la gran mayoría de las etiquetas mantuvieron un rendimiento robusto, con F1-Scores superiores a 0.80, lo que valida la capacidad del modelo para generalizar en el resto del corpus. Sin embargo, la etiqueta **“Implementación”** presentó un desempeño

Tabla 8: Rendimiento de las configuraciones óptimas sobre el conjunto de pruebas, ordenado según la definición temática original de las etiquetas.

Etiqueta	**Precisión**	**Exhaustividad**	**F1-Score**	**Soporte**
Motivación	0.7500	0.5455	0.6316	11
Carga académica	1.0000	1.0000	1.0000	4
Criterios de aceptación	1.0000	0.4545	0.6250	11
Reporte de estado	0.7778	0.9333	0.8485	15
Soporte	0.7143	0.8333	0.7692	6
Comunicación	0.6154	1.0000	0.7619	8
Cara a cara	1.0000	1.0000	1.0000	2
Reuniones	0.7273	1.0000	0.8421	16
Relaciones	0.8750	1.0000	0.9333	7
Profesores	1.0000	1.0000	1.0000	1
Clientes	1.0000	1.0000	1.0000	4
Diseño	0.5556	0.8333	0.6667	6
Ritmo	0.4286	0.8182	0.5625	11
Asignación de tareas	0.8333	0.5556	0.6667	9
Priorización de tareas	0.2500	1.0000	0.4000	1
Alcance	0.5000	0.7778	0.6087	9
Frecuencia de integración	0.5000	0.5000	0.5000	2
Experiencia	0.7500	0.3750	0.5000	8
Tecnologías	0.8750	0.8750	0.8750	16
Calidad de software	0.6000	0.9000	0.7200	10
Implementación	0.0000	0.0000	0.0000	4
Integración	1.0000	0.4375	0.6087	16
Macro F1 Score	-	-	**0.7054**	-
Micro F1 Score	-	-	**0.7209**	-

deficiente, con una precisión de apenas 0.40 y un F1-Score de 0.57. Estos resultados indican que el modelo tiende a sobre-predecir esta clase (alto número de Falsos Positivos), confundiendo posiblemente aspectos generales de la ejecución del proyecto con discusiones específicas sobre implementación de código, la cuales son de interés para esta investigación.

Considerando este bajo rendimiento en el test de sanidad, sumado a los resultados subóptimos observados previamente en la fase de evaluación sobre el conjunto de prueba (donde alcanzó un F1-Score de 0.00), se tomó la decisión metodológica de excluir la etiqueta “Implementación” de los análisis posteriores.

Tabla 9: Resultados de la validación secundaria de sanidad sobre las predicciones del modelo.

Tema	Precisión	Recall	F1-Score
Carga académica	0.80	1.00	0.89
Criterios de aceptación	1.00	1.00	1.00
Clientes	1.00	1.00	1.00
Comunicación	0.70	1.00	0.82
Diseño	0.90	1.00	0.95
Experiencia técnica	0.80	1.00	0.89
Cara a cara	0.70	1.00	0.82
Frecuencia de integración	0.80	1.00	0.89
Implementación	**0.40**	**1.00**	**0.57**
Integración	0.90	1.00	0.95
Reuniones	0.90	1.00	0.95
Motivación	0.90	1.00	0.95
Ritmo	0.70	1.00	0.82
Relaciones	1.00	1.00	1.00
Alcance	0.80	0.80	0.80
Calidad de software	0.80	1.00	0.89
Reporte de estado	0.80	0.80	0.80
Apoyo	0.80	1.00	0.89
Asignación de tareas	0.90	1.00	0.95
Priorización de tareas	0.60	1.00	0.75
Docentes	1.00	1.00	1.00
Tecnologías	1.00	0.91	0.95

4.3.3. Análisis de Sentimiento

De manera complementaria y como tarea auxiliar al etiquetado temático, se implementó un proceso de clasificación de sentimiento utilizando el modelo **GPT-4o**. El objetivo fundamental de esta etapa fue añadir una capa de discernimiento sobre el contenido de las frases, permitiendo interpretar no solo el tema tratado, sino la polaridad de la experiencia descrita. Esto resulta crucial para distinguir entre los éxitos consolidados (aspectos positivos) y las fricciones o desafíos (aspectos negativos) dentro de cada tema y tópico identificado.

El procedimiento se llevó a cabo mediante el uso de la API de pago de OpenAI, procesando iterativamente la totalidad del corpus de frases validadas (2219 en total). Se le solicitó al modelo, frase por frase, que clasificara el texto en una de tres categorías excluyentes: Positivo, Neutro o Negativo. Esta clasificación automática permitió posteriormente cruzar la dimensión emocional con la dimensión temática, facilitando la identificación de contextos de dolor o celebración en las retrospectivas de los estudiantes.

Los resultados de la categorización de las frases en los sentimientos descritos resultó en 980 (44.2 %) frases positivas, 732 (33 %) neutras y 507 (22.9 %) frases con connotación negativa.

4.3.4. Resumen de la Metodología

La ejecución de la metodología propuesta culmina en la transformación de las frases crudas de las retrospectivas en entidades de información estructurada y enriquecida. Como se detalló en las secciones previas, este proceso integra la capacidad de generalización del LLM (mediante la selección de *prompts* óptimos) con la capacidad de descubrimiento de patrones del modelado de tópicos.

La Figura 4 presenta el modelo conceptual de los datos resultantes. En este esquema, la entidad central es la **Frase**, la cual conserva sus atributos de origen: contenido textual, Sprint, equipo y la sección de la retrospectiva a la que pertenece (ej. “Qué salió bien” o “Qué salió mal”). Esta entidad base se enriquece mediante tres dimensiones de análisis interconectadas:

1. **Etiquetas:** La frase se vincula con una o más categorías provenientes de la taxonomía de 21 definiciones, asignadas por el modelo GPT-OSS-20B.
2. **Tópico:** La frase se asocia a un clúster semántico específico descubierto por el algoritmo BERTopic, aportando un contexto situacional detallado.
3. **Sentimiento:** Se incorpora un atributo de polaridad (Positivo, Neutro o Negativo) generado por el modelo GPT-4o, lo que permite discernir la carga emocional y el contexto de éxito o fricción de la oración.

Cabe mencionar que, dada la naturaleza de los métodos, puede ocurrir que una frase no tenga asignada ninguna etiqueta o no esté asociada a un tópico (si es considerada “ruido” por el algoritmo de agrupamiento HDBSCAN). Sin embargo, el atributo de sentimiento busca ser transversal a todas las instancias procesadas.

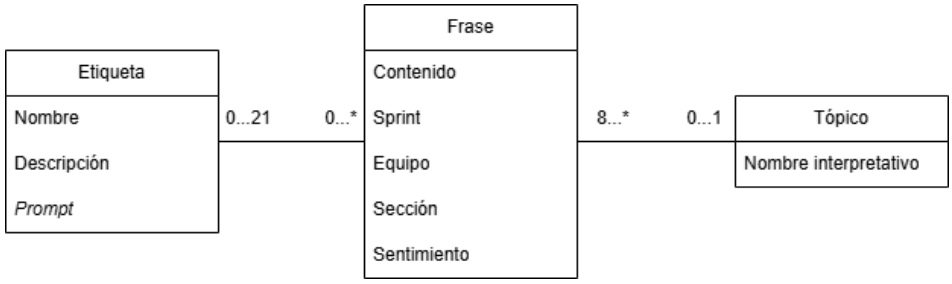


Figura 3: Diagrama Entidad-Relación que modela la estructura final de los datos procesados, destacando la inclusión de la sección de origen junto a las dimensiones semánticas.

La Figura 4 ilustra esta dinámica de intersección. En el ejemplo, se observa cómo una etiqueta general como “Reuniones” (con 150 frases detectadas) no es un bloque monolítico, sino que se distribuye hacia tópicos específicos como “Realizar reuniones todos los días” o “El papel

de las tecnologías en las reuniones". De igual forma, un mismo tópico puede nutrirse de frases que fueron clasificadas bajo distintas etiquetas (como el cruce entre "Tecnologías" y "Reuniones"), revelando la naturaleza multidimensional de las discusiones estudiantiles.

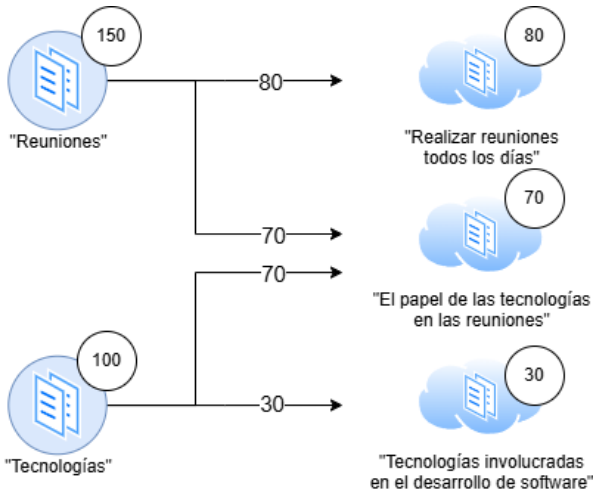


Figura 4: Ejemplo de la información que aportan los Temas y Tópicos a cada frase.

Esta estructura multidimensional permite un análisis de los resultados con distintos niveles de granularidad. Mientras que las etiquetas ofrecen una visión macroscópica y estandarizada de los temas, los tópicos desagregan estos conceptos en discusiones concretas. A su vez, el cruce entre el sentimiento detectado y la sección de origen permite validar la coherencia del discurso (ej. verificar si lo escrito en "Qué salió mal" tiene efectivamente una carga negativa) y cualificar dichas discusiones como problemáticas o exitosas con mayor precisión.

En consecuencia, la sección de resultados que se presenta a continuación no solo reporta métricas de frecuencia, sino que explotará esta relación para responder no solo *qué* temas se discutieron (Etiquetas), sino *en qué contexto* y *con qué matices* (Tópicos y Sentimientos) fueron abordados por los estudiantes a lo largo del curso *capstone*, dando una respuesta adecuada a las preguntas de investigación descritas.

Finalmente, con el objetivo de garantizar la transparencia, reproducibilidad y auditabilidad de esta investigación, se pone a disposición el conjunto de datos completo generado durante el estudio. Este repositorio incluye: (1) el listado detallado de los tópicos identificados por BERTopic, (2) las etiquetas temáticas predichas por el modelo LLM para la totalidad del corpus, y (3) los conjuntos de frases validados bajo consenso de expertos para cada una de las etiquetas temáticas. Estos recursos se encuentran alojados y accesibles en el enlace digital del autor de la tesis[3].

[3] https://drive.google.com/drive/folders/11F4dohEVvnHCVANdIa0QJBHpcAf8s3Al?usp=drive_link

CAPÍTULO 5
RESULTADOS Y DISCUSIÓN

A continuación se presentan los hallazgos derivados del procesamiento y análisis de las 2219 frases extraídas de las retrospectivas de Sprint. Antes de profundizar en los Temas específicos, se estableció el alcance, la representatividad y el comportamiento demográfico del análisis logrado mediante la metodología híbrida propuesta. Esta validación inicial es necesaria para asegurar que las conclusiones extraídas provienen de una visión comprensiva de la realidad estudiantil y para contextualizar la densidad de la información procesada.

La Figura 5 ilustra la cobertura del proceso de clasificación y modelado de tópicos. Se observa que la combinación del etiquetado deductivo (LLM) y el modelado inductivo (Tópicos) logra abarcar un **95.9 %** del contenido total disponible.

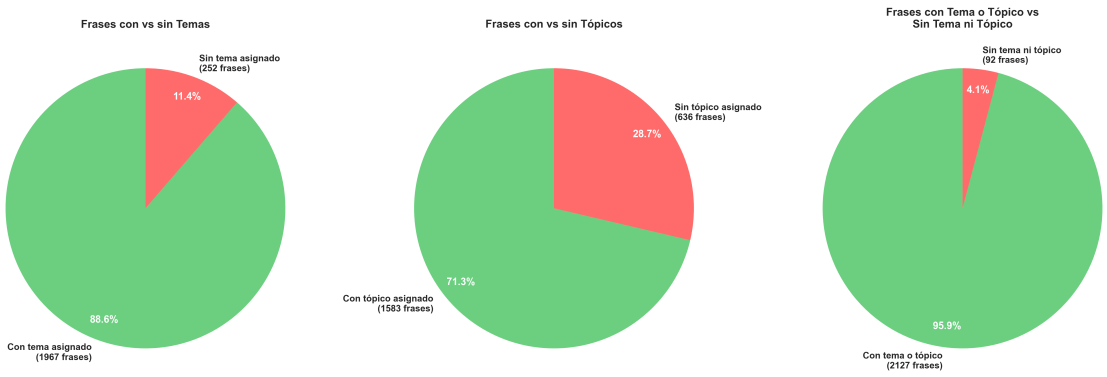


Figura 5: Distribución de frases según la asignación de Temas y Tópicos.

El desglose de la Figura 5 revela que el enfoque basado en etiquetas predefinidas (Temas) logró clasificar exitosamente el 88.6 % de las frases, proporcionando una base sólida para el análisis estructurado. Por su parte, el modelado de tópicos aportó contexto granular al 71.3 % de los datos. La unión de ambos conjuntos confirma que la metodología es robusta para ofrecer un diagnóstico fiel de la experiencia capstone, minimizando la información no observada.

Una vez validada la cobertura, es preciso examinar cómo se distribuye este volumen de información a través de las variables temporales y estructurales del estudio. Las Figuras 6, 7 y 8 presentan la distribución de la cantidad de frases generadas según el año académico, el Sprint y la sección de la retrospectiva, respectivamente.

Al analizar la Figura 6, se detecta un comportamiento relativamente estable entre los años 2022 y 2023, con medianas cercanas a las 12 frases por equipo. Sin embargo, el año 2024 muestra un cambio notable en la dinámica de documentación: no solo aumenta la media de frases recolectadas (representada por el triángulo verde), sino que la dispersión de los datos

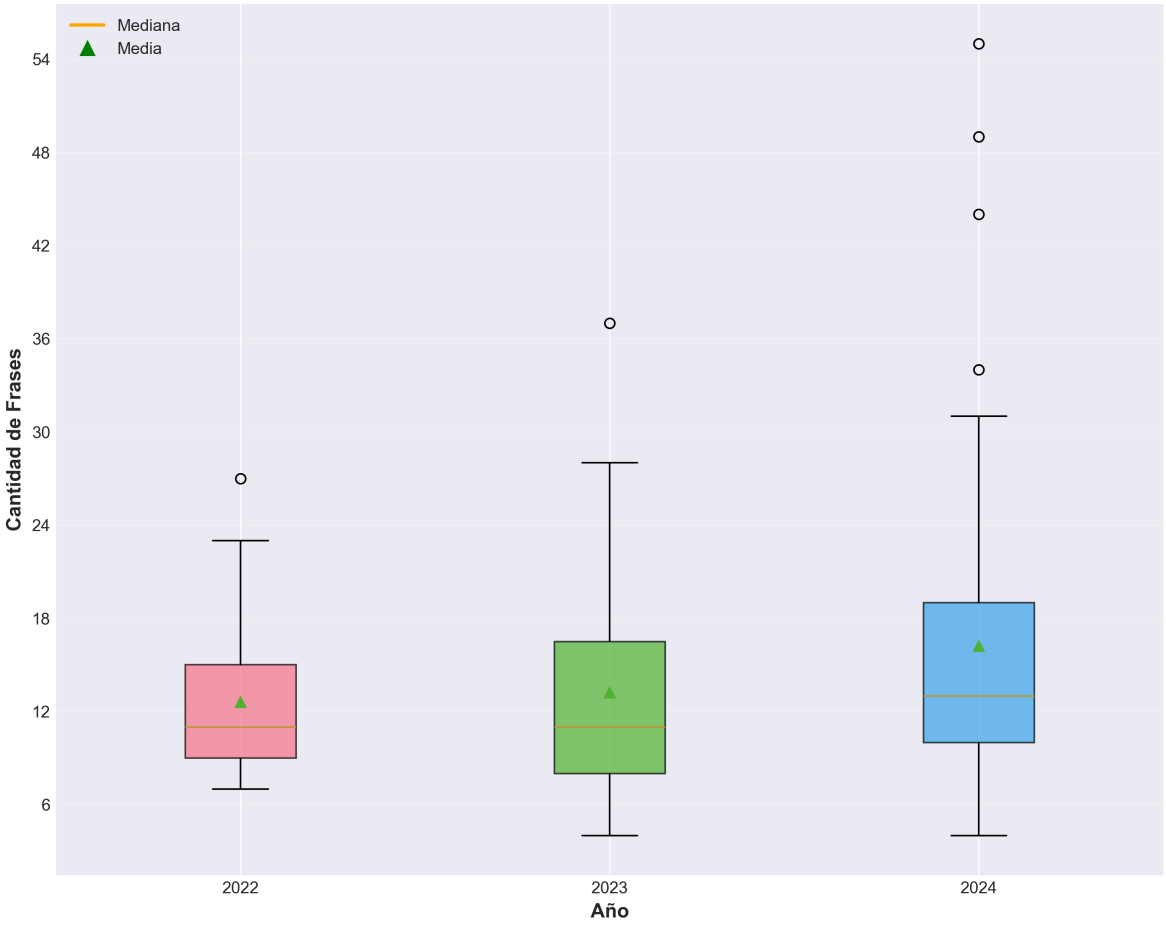


Figura 6: Cantidad de frases identificadas por año académico.

crece significativamente. La presencia de valores atípicos (*outliers*) en la parte superior del gráfico de 2024 indica que hubo equipos particularmente detallistas en sus retrospectivas, generando más de 40 o 50 frases procesables. Esto sugiere que la variabilidad en la expresividad y el compromiso documental de los estudiantes puede fluctuar considerablemente entre cohortes.

La Figura 7 revela una tendencia decreciente en la cantidad de información generada conforme avanza el semestre. El Sprint 0 presenta la mayor mediana y dispersión, lo cual es coherente con la etapa inicial de formación de equipos, donde la incertidumbre y la necesidad de acuerdos generan discusiones más extensas. Ahora bien, sin considerar este Sprint que concentra a menos equipos, a partir del Sprint 1 en adelante, la mediana de frases desciende progresivamente, lo que podría interpretarse como un signo de eficiencia en equipos maduros que requieren menos palabras para comunicarse, o como un síntoma de fatiga documental hacia el final del periodo académico.

Finalmente, la Figura 8 ofrece una perspectiva crítica sobre la estructura de la reflexión. Es importante notar una distinción metodológica en los datos: las secciones orientadas explíci-

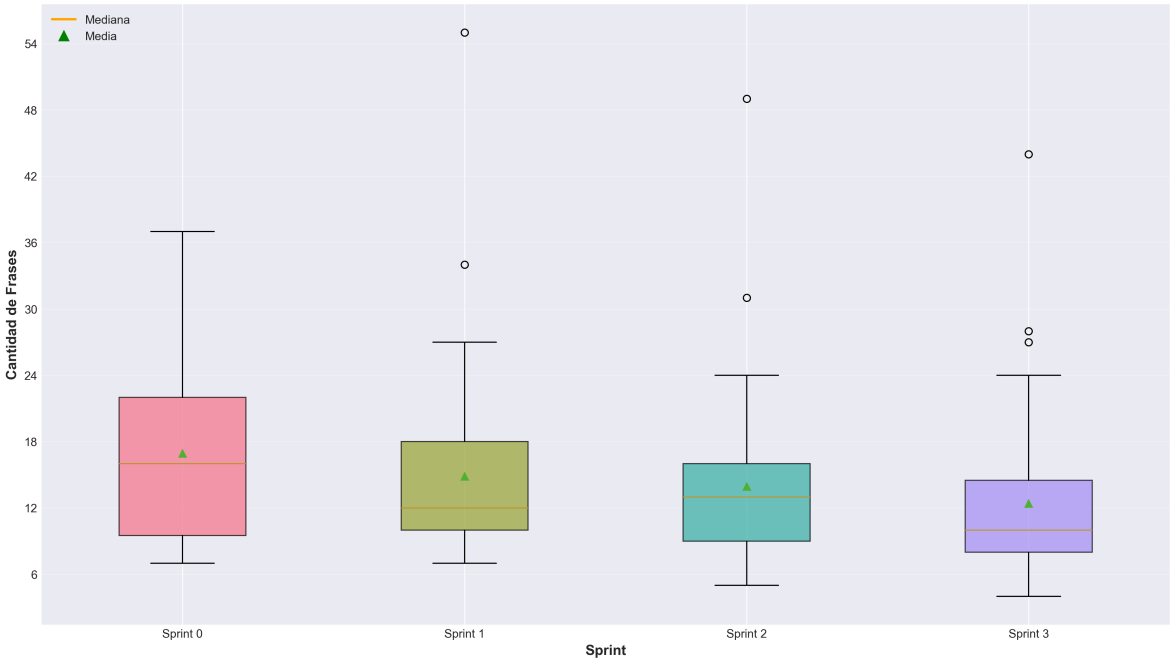


Figura 7: Evolución de la cantidad de frases por Sprint.

tamente a la modificación de hábitos (“¿Qué debería empezar a hacer?” y “¿Qué se debería dejar de hacer?”) representan casos atípicos o formatos específicos utilizados por una minoría de equipos, habiendo 41 frases (1.8 % del total) en la primera y 6 frases (0.2 % del total) en la última, por lo que la distribución de frases que se muestra de dichas secciones no es generalizable al curso completo.

El análisis sustancial recae en las tres secciones canónicas: “¿Qué salió bien?”, “¿Qué salió mal?” y “¿Qué se debería mantener?”. Al observar estas categorías, se confirma la tendencia hacia el refuerzo positivo detectada anteriormente: el volumen de frases en “¿Qué salió bien?” (819 frases) y “¿Qué mantener?” (720 frases) supera a la reflexión sobre los fallos (“¿Qué salió mal?”, 633 frases). Asimismo, la presencia de numerosos valores atípicos en estas secciones principales indica que la expresividad es altamente variable; mientras la mayoría de los equipos mantiene un registro conciso (medianas entre 4 y 5 frases por grupo), existen grupos que documentan exhaustivamente cada sesión, elevando significativamente los promedios y generando la dispersión observada.

5.1. Análisis de la presencia de Temas dentro de las Retrospectivas de Sprint

Habiendo establecido la representatividad de los datos, abordamos la primera pregunta de investigación (**RQ1**), referente a la frecuencia y cobertura de los Temas. Este análisis permite

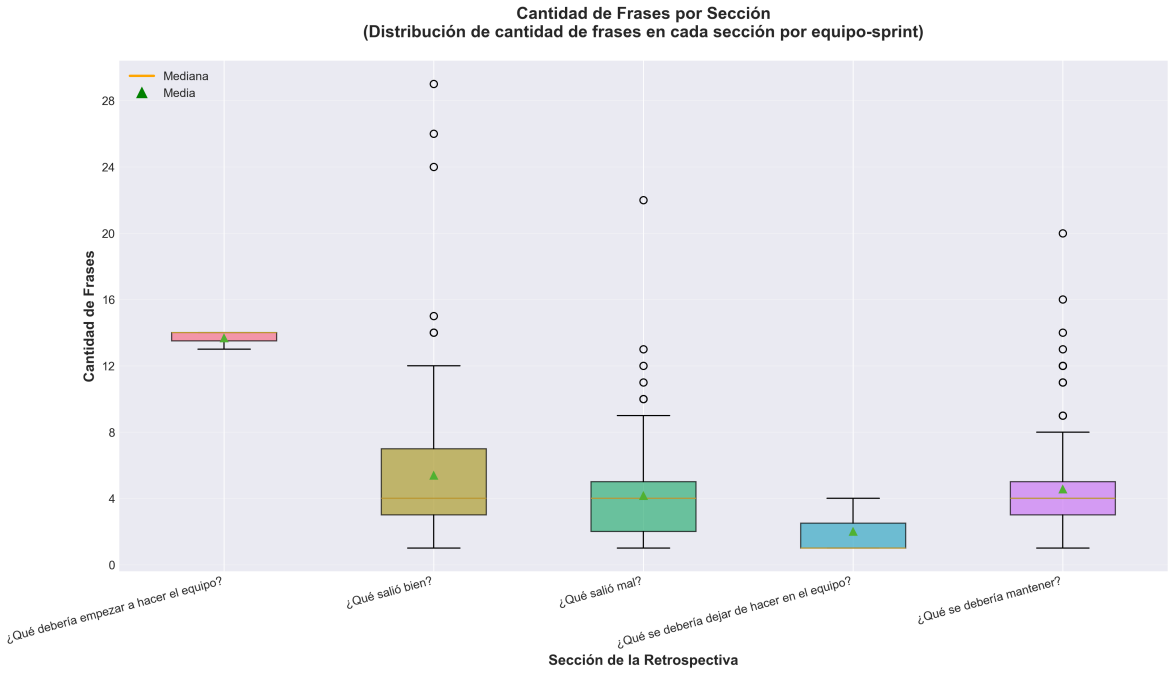


Figura 8: Distribución de frases por sección de la retrospectiva.

jerarquizar las preocupaciones de los equipos, revelando qué aspectos dominan la conversación y cuáles pasan a un segundo plano.

La Figura 9 muestra la distribución de frecuencia para las etiquetas temáticas en el conjunto de datos completo.

Al analizar la Figura 9, destaca inmediatamente que la reflexión estudiantil presenta una fuerte concentración en aspectos operativos y de gestión interna. Las etiquetas **Ritmo**, **Reuniones** y **Alcance** lideran el ranking con una diferencia notable respecto al resto, acumulando gran parte de la discusión. Esto sugiere que, para los estudiantes en este contexto, el desafío principal percibido no es necesariamente técnico o de ingeniería compleja, sino organizativo: cómo mantener una velocidad de trabajo constante, cómo sincronizarse efectivamente y cómo manejar y cumplir la cantidad de trabajo comprometido.

En un segundo nivel de frecuencia, encontramos un bloque de Temas “habilitadores” como **Tecnologías**, **Motivación**, **Comunicación** y **Reporte de estado**. Es interesante notar que la dimensión humana (*Motivación*, *Comunicación*) tiene un peso comparable a la dimensión técnica instrumental (*Tecnologías*), lo que refuerza la naturaleza socio-técnica de estos proyectos. Por el contrario, etiquetas relacionadas con actores externos, como **Clientes** y **Profesores**, aparecen en el cuartil inferior de frecuencia, indicando una tendencia de los equipos estudiados a volcarse hacia adentro durante las retrospectivas. Esto es esperable, pero la poca discusión relacionada con la actitud y dinámica del equipo con el cliente sugiere que esto se considera también un aspecto menos relevante, lo que se contradice con la relevancia de este aspecto en los proyectos de desarrollo de software en la práctica y es indeseado para la

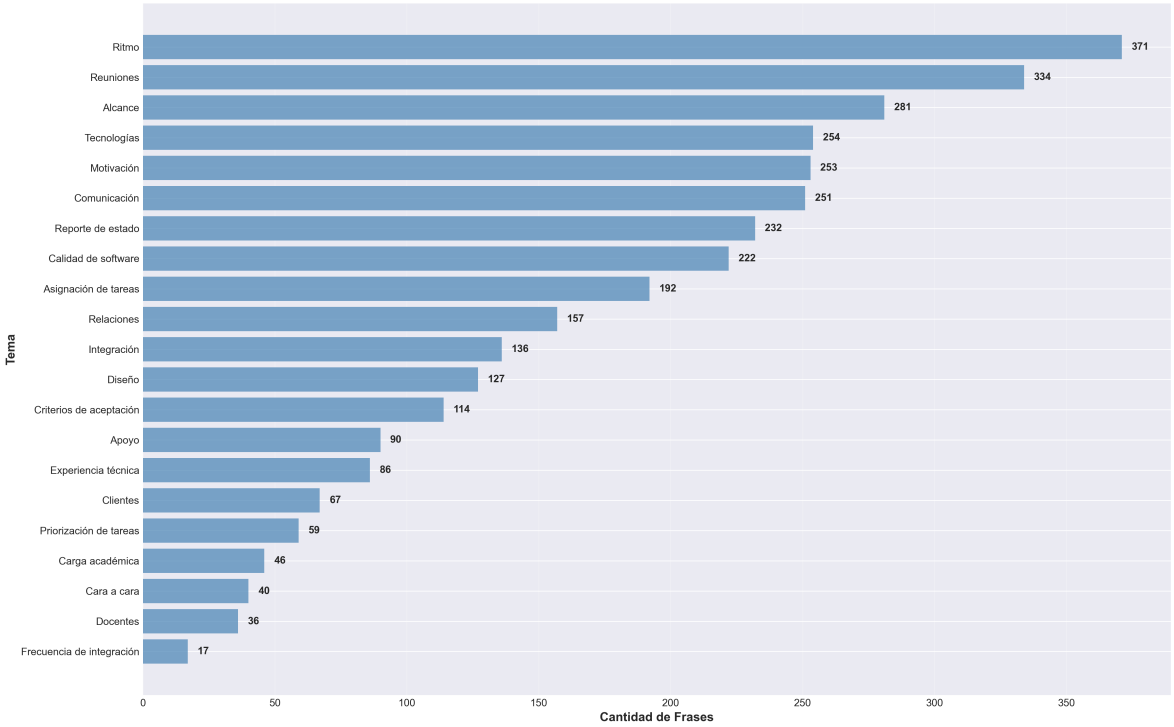


Figura 9: Distribución de frecuencia total de frases por etiqueta temática en todas las frases.

formación de los estudiantes.

Para validar si esta distribución es estructural más allá de una cohorte específica, se observa el comportamiento de estos Temas a través de los diferentes años estudiados en la Figura 10.

La Figura 10 confirma la robustez de los hallazgos a lo largo de los años 2022, 2023 y 2024. Independientemente de la cohorte, los estudiantes reproducen los mismos patrones de discusión: una alta carga cognitiva dedicada a regular la dinámica interna y una atención menor a Temas como la **Calidad de software** o los **Criterios de aceptación**. Esta invarianza temporal sugiere que los resultados reflejan la realidad estructural de la experiencia *capstone*, donde el desafío principal radica en la transición hacia un flujo de trabajo ágil y colaborativo.

Para complementar la visión cuantitativa de la frecuencia, es imperativo analizar la cualidad de estas discusiones. La Figura 11 desglosa la polaridad de las frases (Positiva, Neutra, Negativa) para cada etiqueta temática, mientras que la Figura 12 contextualiza estas menciones según la pregunta de la retrospectiva que las originó.

Por lo que se puede apreciar, los mayores dolores de los equipos se concentran en la gestión del flujo de trabajo y la planificación. Los Temas **Ritmo** y **Alcance** encabezan indiscutiblemente la carga negativa y dominan las secciones de “¿Qué salió mal?” y “¿Qué se debería dejar de hacer?”. Esta configuración sugiere que la estimación del esfuerzo y la constancia en la

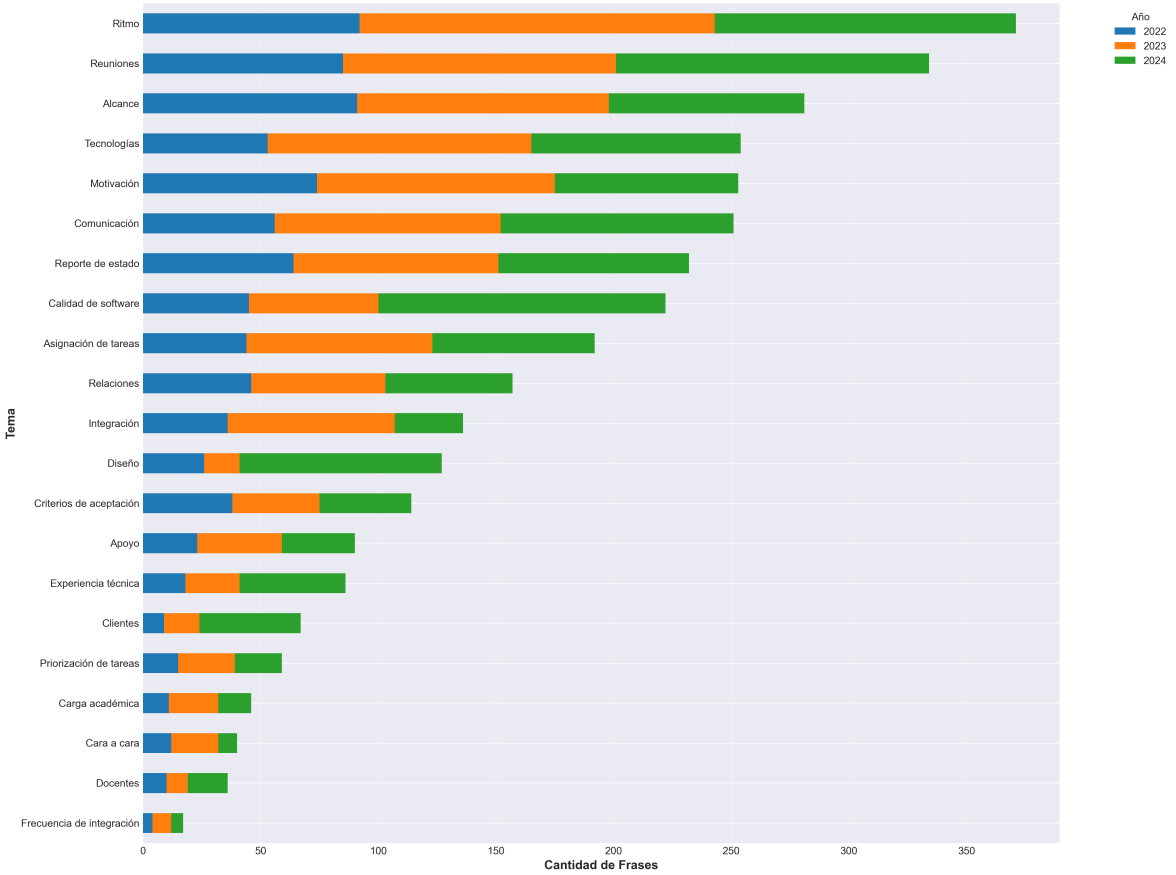


Figura 10: Frecuencia de Temas agrupada por año.

velocidad de trabajo son las fuentes de frustración más agudas y persistentes; los estudiantes perciben recurrentemente que avanzan más lento de lo planeado o que se comprometen a un alcance que no logran completar. En una línea similar, la **Asignación de tareas** y la **Priorización de tareas** tienden hacia la neutralidad y lo negativo, respectivamente, reflejando fricciones operativas al momento de distribuir equitativamente el trabajo o decidir qué es urgente.

En el extremo opuesto, la dimensión humana y colaborativa emerge como el bastión de la resiliencia del equipo. Las etiquetas **Relaciones**, **Motivación** y **Apoyo** presentan un sentimiento predominantemente positivo y dominan las secciones de “¿Qué salió bien?”. Esto evidencia que, aun cuando el proyecto técnico o la gestión fallen, la cohesión interpersonal y el buen trato actúan como amortiguadores del estrés. De manera interesante, las **Reuniones**, a pesar de su altísima frecuencia, comparten esta polaridad positiva y tienen una fuerte presencia en “¿Qué mantener?”. Lejos de ser percibidas como una carga burocrática, los estudiantes valoran estos espacios de sincronización.

La dimensión técnica presenta un comportamiento más complejo y mixto. El Tema **Tecnologías** tiene una base positiva considerable, aludiendo a herramientas que facilitan el trabajo,

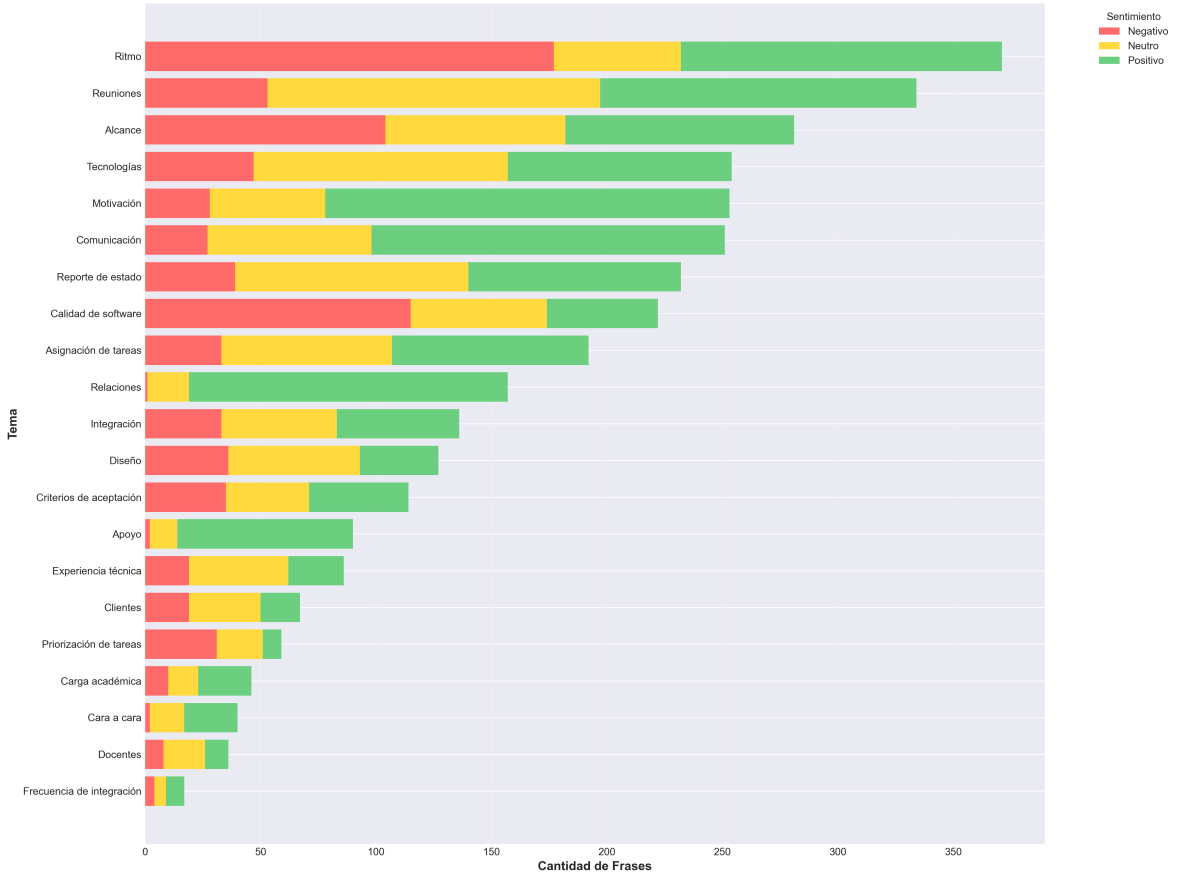


Figura 11: Distribución de sentimientos por etiqueta temática.

pero su presencia notable en secciones de “¿Qué empezar a hacer?” refleja que la tecnología es vista dinámicamente como un área de mejora continua. Por otro lado, la **Calidad de Software**, los **Criterios de Aceptación** y la **Integración** muestran cargas negativas relevantes, sugiriendo que la deuda técnica y los conflictos de código son dolores conscientes que intentan mitigar, aunque a menudo sean desplazados por la urgencia del ritmo.

Finalmente, los factores externos a los equipos muestran impactos polarizados. Las interacciones con **Clientes** y **Docentes** son marginales en volumen y expresadas tanto positiva como negativamente. Por otro lado, la **Carga Académica** es un factor casi exclusivamente negativo que impacta el rendimiento, mostrándose como un obstáculo y no como un desafío superado. En contraste, la presencialidad (**Cara a cara**) se valora positivamente como un catalizador de la productividad.

La triangulación de los análisis permite decir que la **autogestión y la operatividad** constituyen el núcleo de la dificultad del curso, siendo consistentemente negativas y transversales. En contraparte, la **dimensión social y humana** emerge como el principal activo, proporcionando el soporte psicológico necesario ante la frustración metodológica.

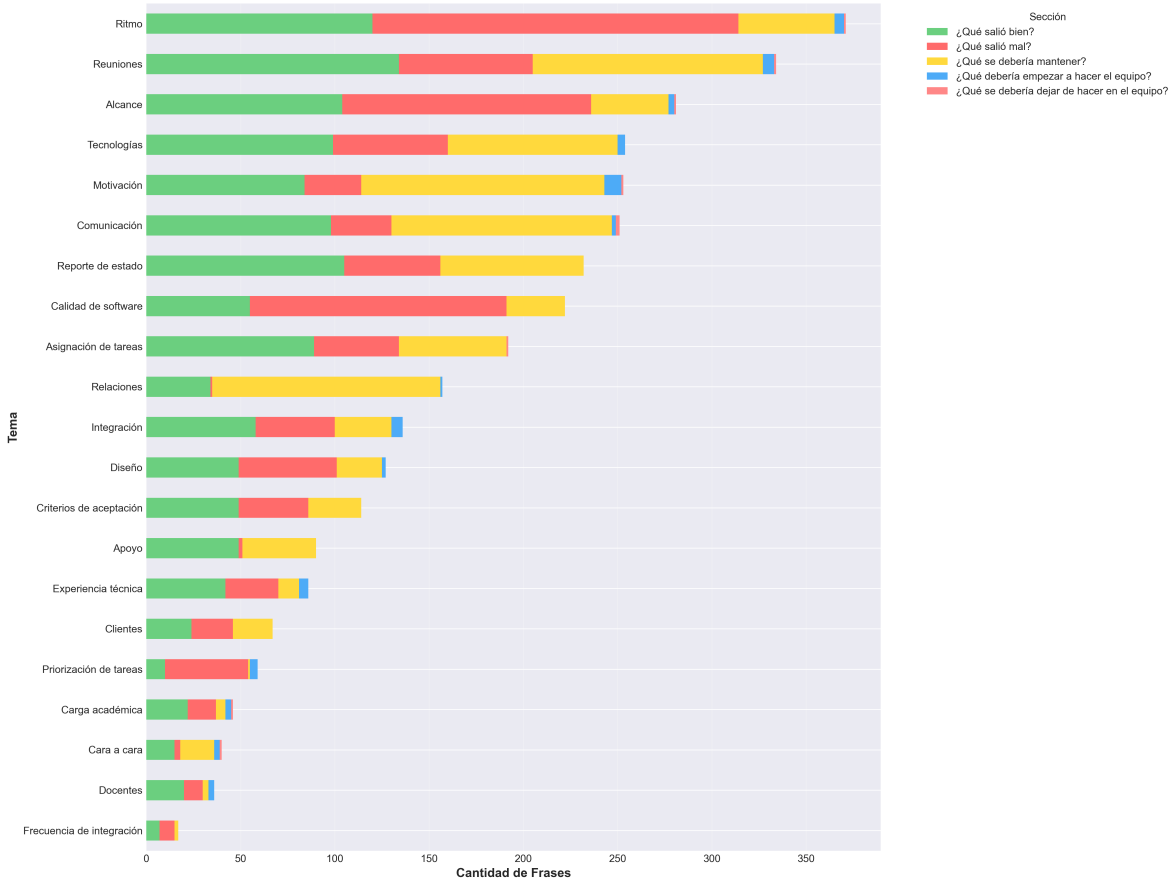


Figura 12: Distribución de frases según la sección de la retrospectiva en la que fueron mencionadas.

Por otro lado, para profundizar en la discusión que permita dar una respuesta a la pregunta de investigación **RQ3** ("¿Cómo evoluciona la reflexión de los equipos…?"), es crucial entender cómo cambia el *sentimiento* y la presencia de estas discusiones a medida que el proyecto madura Sprint a Sprint. La Figura 13 desglosa detalladamente la evolución del sentimiento por Sprint para la totalidad de los Temas identificados.

Al examinar estas trayectorias detalladas en conjunto, se confirman y matizan los patrones evolutivos del curso, permitiendo identificar comportamientos distintos en la maduración de los equipos:

- **Persistencia de la Fricción Operativa:** Se observa claramente que el Tema **Ritmo** mantiene una barra roja (negativa) predominante y constante desde el Sprint 0 hasta el Sprint 3, acompañada siempre de una porción amarilla (neutra) y verde (positiva) menores. Esto demuestra que la dificultad para mantener la velocidad no es un problema de ajuste inicial que se resuelve con el tiempo, sino una condición crónica del proyecto que acompaña a los equipos hasta el final. Similarmente, el **Alcance** muestra picos de negatividad en los Sprints 1 y 2, momentos críticos donde la realidad del desarrollo

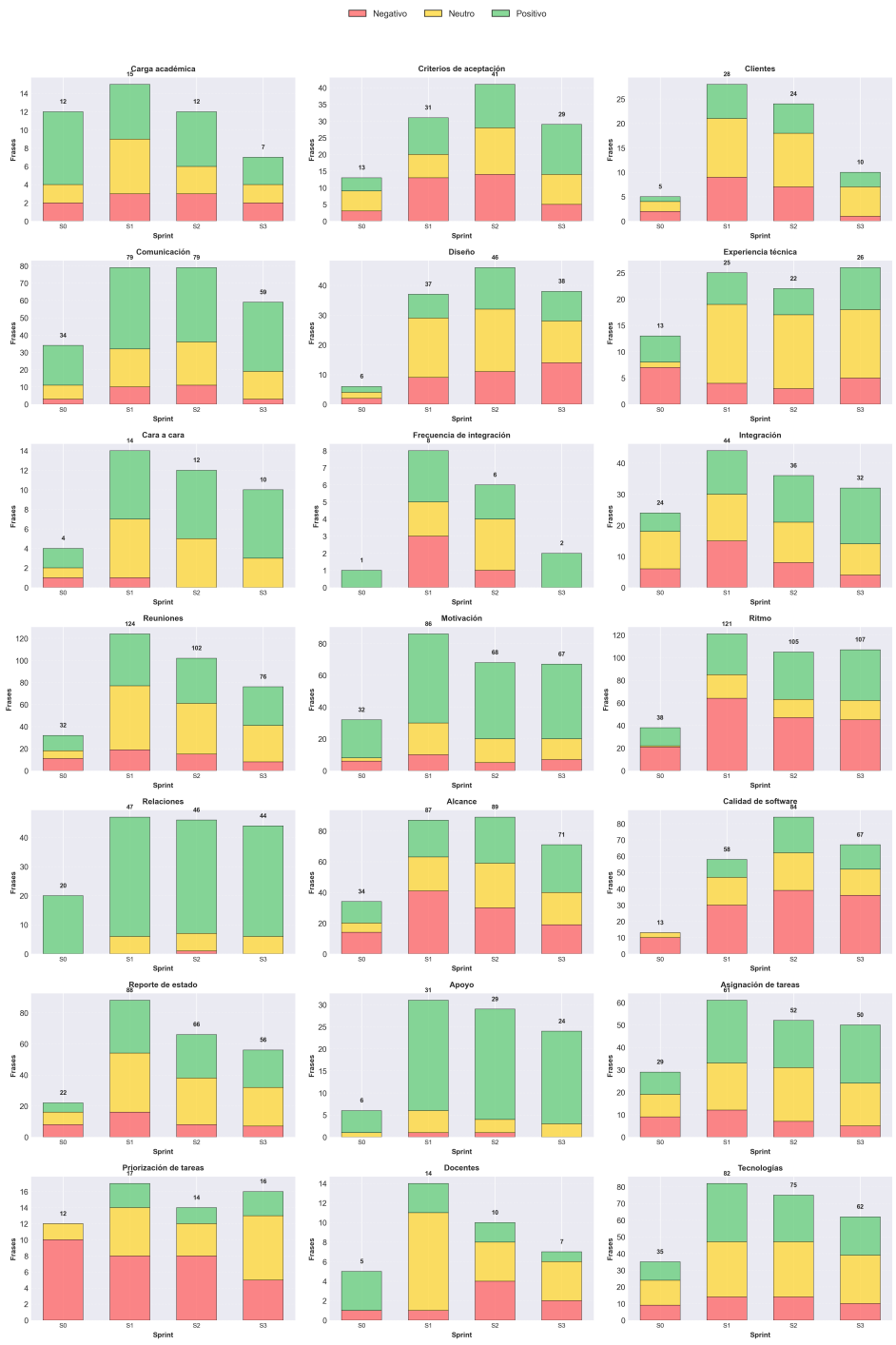


Figura 13: Evolución de Temas por Sprint - Distribución por Sentimiento.

choca con las estimaciones iniciales, obligando a reajustes dolorosos.

- **Resiliencia Social Sostenida:** En contraste, las gráficas de **Motivación**, **Relaciones** y **Apoyo** exhiben barras verdes robustas y estables a lo largo de todo el ciclo. Incluso

en el Sprint 3, cuando la presión de entrega es máxima, la proporción positiva de estos Temas no decae significativamente. Esto sugiere un hallazgo clave: el capital social construido por los equipos no se erosiona linealmente con el estrés del proyecto, sino que se mantiene como un recurso renovable que sostiene la moral. La **Comunicación** también muestra este patrón de positividad sostenida, aunque con un leve aumento de menciones negativas hacia el final, probablemente asociadas a la coordinación del cierre.

- **Maduración Técnica y Deuda Visible:** La visualización revela una evolución interesante en lo técnico. El Tema **Tecnologías** muestra un crecimiento en volumen hacia los Sprints intermedios (S1 y S2), manteniendo una base positiva sólida, lo que indica una fase de adopción y dominio de herramientas. Sin embargo, Temas como **Diseño** y **Calidad de Software** incrementan su presencia y su componente negativo/mixto hacia el final (Sprints 2 y 3). Esto refleja un patrón de madurez realista: la deuda técnica (diseño descuidado, falta de pruebas) suele permanecer oculta al inicio y se hace visible y dolorosa solo cuando el sistema alcanza cierta complejidad, obligando a los equipos a discutir sobre refactorización en las etapas tardías.
- **Ajuste Metodológico Temprano:** Temas como **Criterios de aceptación** y **Priorización de tareas** muestran una actividad relevante en los primeros Sprints (S0, S1) con componentes negativos importantes, para luego estabilizarse o disminuir relativamente en impacto negativo. Esto sugiere que los equipos sufren el choque metodológico al principio, aprendiendo a calibrar sus definiciones de "terminado", sus prioridades y la manera de trabajar para lograr el comportamiento deseado en las funcionalidades que desarrollan en la primera mitad del proyecto.

En síntesis, la evolución de la reflexión no es lineal hacia la mejora absoluta en todos los frentes. Mientras los equipos logran estabilizar tempranamente sus dinámicas humanas (éxito social sostenido), luchan permanentemente con la gestión del tiempo (fricción operativa constante) y enfrentan desafíos de calidad creciente a medida que el producto escala (conciencia tardía de la deuda técnica).

5.2. Análisis de los Tópicos encontrados en las Retrospectivas de Sprint

Complementando el análisis deductivo anterior, se procedió a examinar los resultados del modelado inductivo generado mediante BERTopic. Cabe recordar que, para esta configuración experimental, se estableció un parámetro de tamaño mínimo de clúster de 8 frases. Esta decisión metodológica fue crucial para filtrar el ruido inherente al lenguaje natural no estructurado, garantizando que cada uno de los 68 Tópicos detectados representara un patrón de discusión recurrente, denso y sustancial, descartando anécdotas aisladas.

Para facilitar la navegación y la interpretación semántica de estos 68 Tópicos (cuyo detalle exhaustivo se encuentra disponible en el Anexo en la Tabla 11), se realizó un proceso de in-

terpretación y etiquetado manual. Este ejercicio permitió agrupar los clústeres dispersos en 5 macro-categorías temáticas, ofreciendo una capa de abstracción intermedia que permite visualizar las grandes dimensiones de la reflexión estudiantil más allá de la granularidad específica de cada micro-Tópico.

La Figura 14 presenta la distribución de frases agrupadas en estas cinco categorías principales, ofreciendo una primera panorámica de los focos de atención.

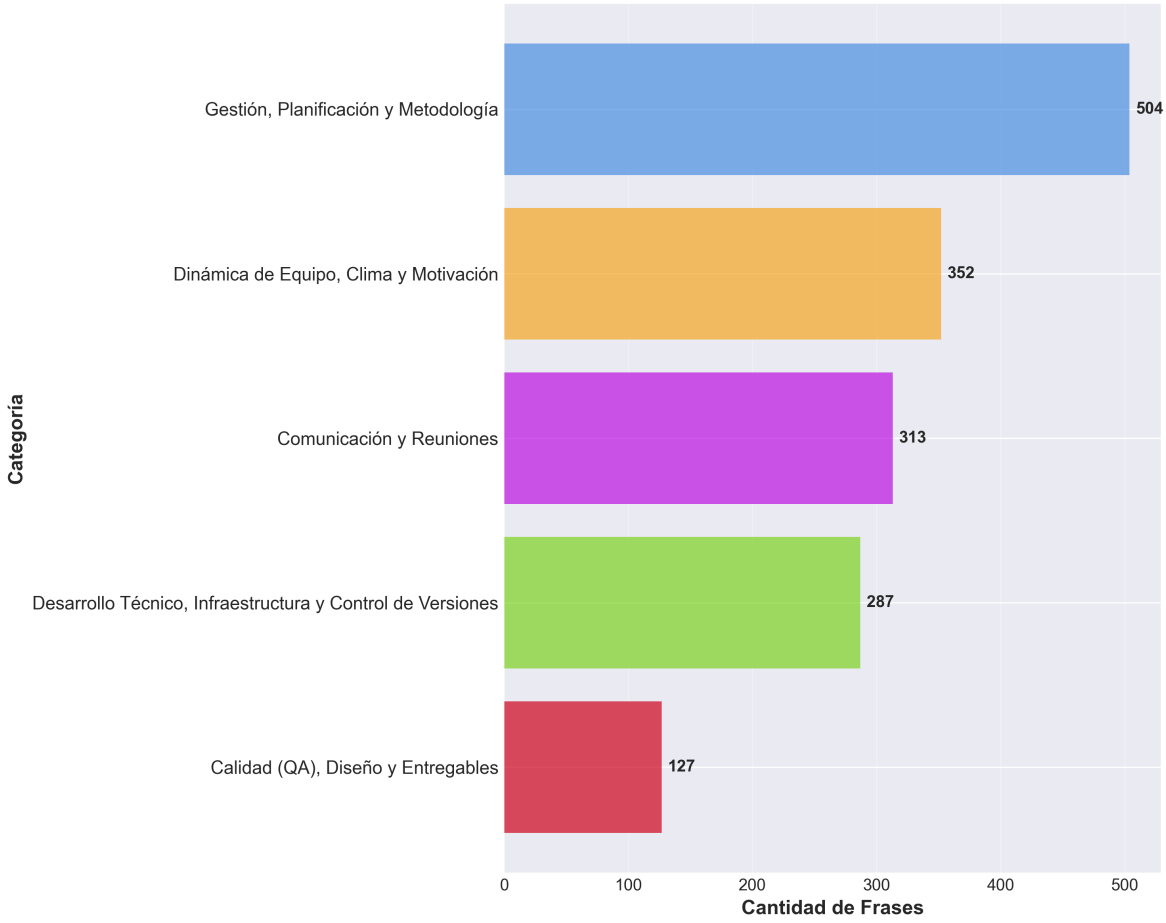


Figura 14: Distribución de frases por Categoría de Tópicos.

El análisis de la Figura 14 entrega una visión reveladora sobre la naturaleza de las preocupaciones en los proyectos *capstone*. Las tres barras superiores —**Gestión, Planificación y Metodología** (504 frases), **Dinámica de Equipo, Clima y Motivación** (352 frases) y **Comunicación y Reuniones** (313 frases)— confirman que el grueso de la discusión plasmada en los resúmenes se enfoca en los aspectos “blandos” y de proceso. Sumadas, estas tres dimensiones acumulan más de 1100 frases, superando ampliamente a las categorías de **Desarrollo Técnico** (287 frases) y **Calidad** (127 frases). Este hallazgo sugiere que la complejidad percibida por los estudiantes no radica tanto en la dificultad técnica de la implementación, sino en la fricción organizativa, la coordinación humana y la disciplina metodológica requerida para

construir software en equipo.

De manera más detallada y respecto a la dimensión del análisis de sentimientos que se aplica a estos hallazgos, en la Figura 15 se puede ver claramente que que los Tópicos descubiertos llevan implícita una carga emocional fuerte en su propia semántica. Esto revela una dicotomía fundamental: la categoría de **Gestión** es la única predominantemente negativa, confirmando que la metodología es la fuente principal de dolor. En contraste, la categoría **Dinámica de Equipo** es un bastión de positividad, demostrando que en las discusiones dentro de las retrospectivas, cuando se reconoce como relevante este aspecto, se hace comúnmente como un éxito para los equipos. La Figura 16 contextualiza este hallazgo: la Gestión domina la sección "¿Qué salió mal?", mientras que la Dinámica de Equipo inunda la sección "¿Qué salió bien?".

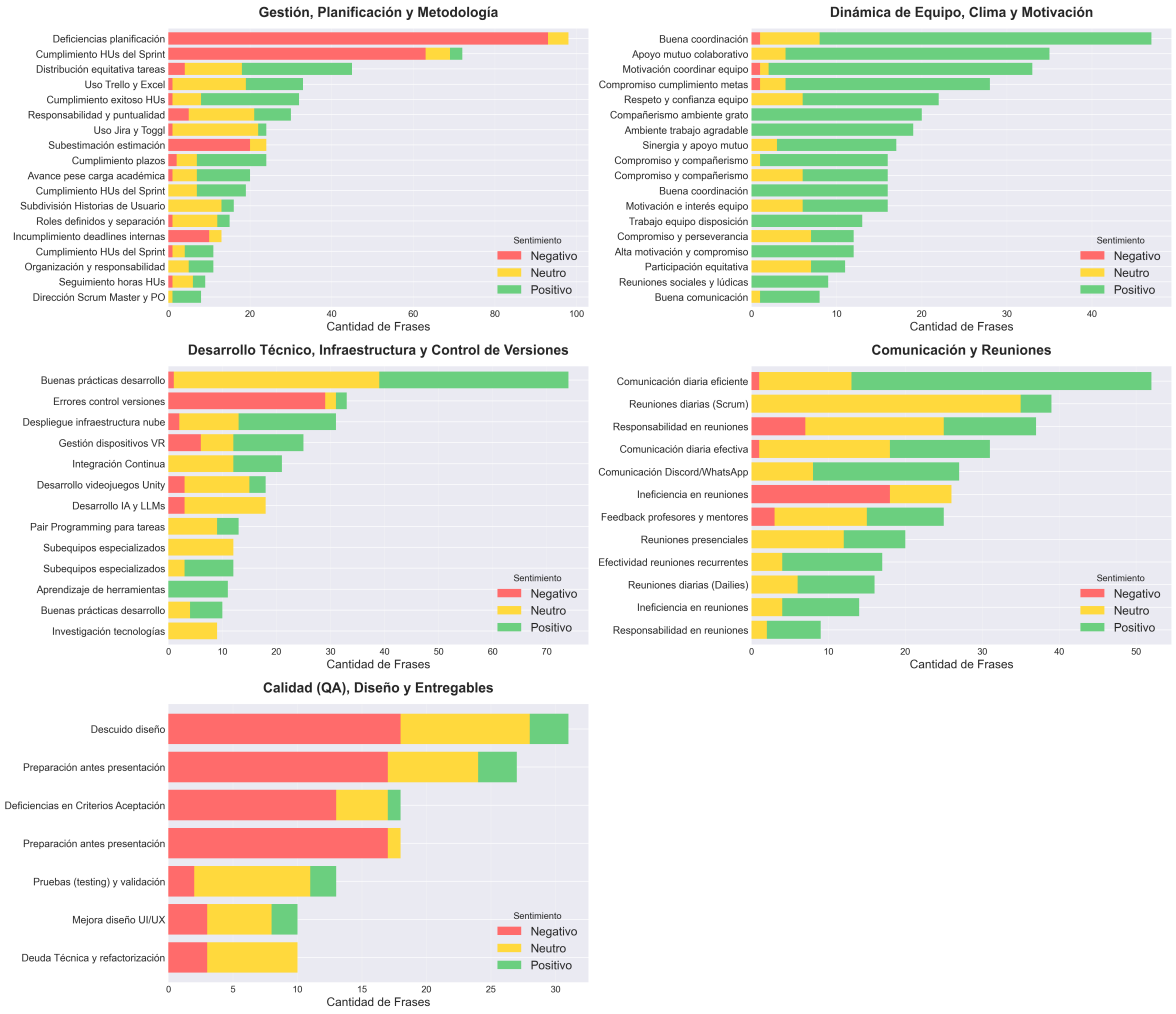


Figura 15: Distribución de frases por Sentimiento asignado y por categoría.

Para cerrar el círculo entre el enfoque deductivo (Temas predefinidos) y el inductivo (Tópicos descubiertos), las siguientes tres figuras presentan una radiografía detallada de la composi-

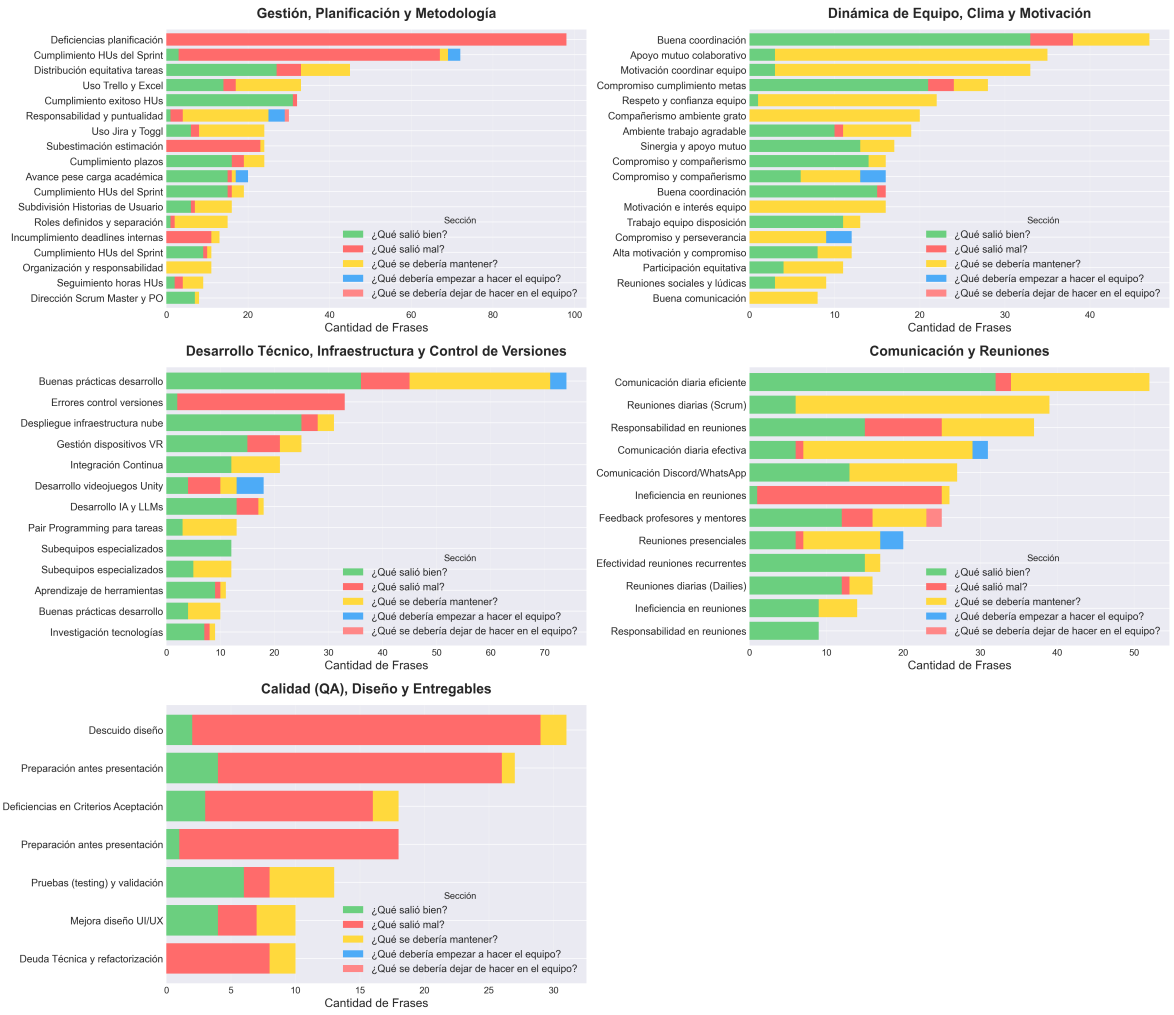


Figura 16: Distribución de frases por Sección de la Retrospectiva y categoría.

ción semántica de cada etiqueta. Este análisis cruzado es esencial para validar la coherencia interna del modelo utilizado para el etiquetado deductivo y permite responder con precisión en qué contexto se discute cada Tema.

La Figura 17 (Parte 1) revela contextos contrastantes. En el Tema **Apoyo**, el Tópico dominante es “Apoyo mutuo colaborativo” (24 %), lo que contextualiza este Tema inequívocamente como un éxito de cohesión grupal. Similarmente, **Comunicación** se define por la “Comunicación diaria efectiva” (18 %), alejándose de la queja y acercándose a la celebración de la fluidez. En la vereda opuesta, el Tema **Alcance** muestra una realidad problemática: aunque existe un 10 % de “Cumplimiento exitoso”, la presencia combinada de “Subestimación” (7 %) y “Deficiencias de planificación” (6 %) confirma que el alcance se discute frecuentemente desde la falla en la predicción. La **Asignación de tareas** se contextualiza en la búsqueda de justicia, dominada por la “Distribución equitativa” (23 %), mientras que **Calidad de software** se asocia preocupantemente al “Descuido de diseño” (11 %). Respecto a los **Clientes**, la alta

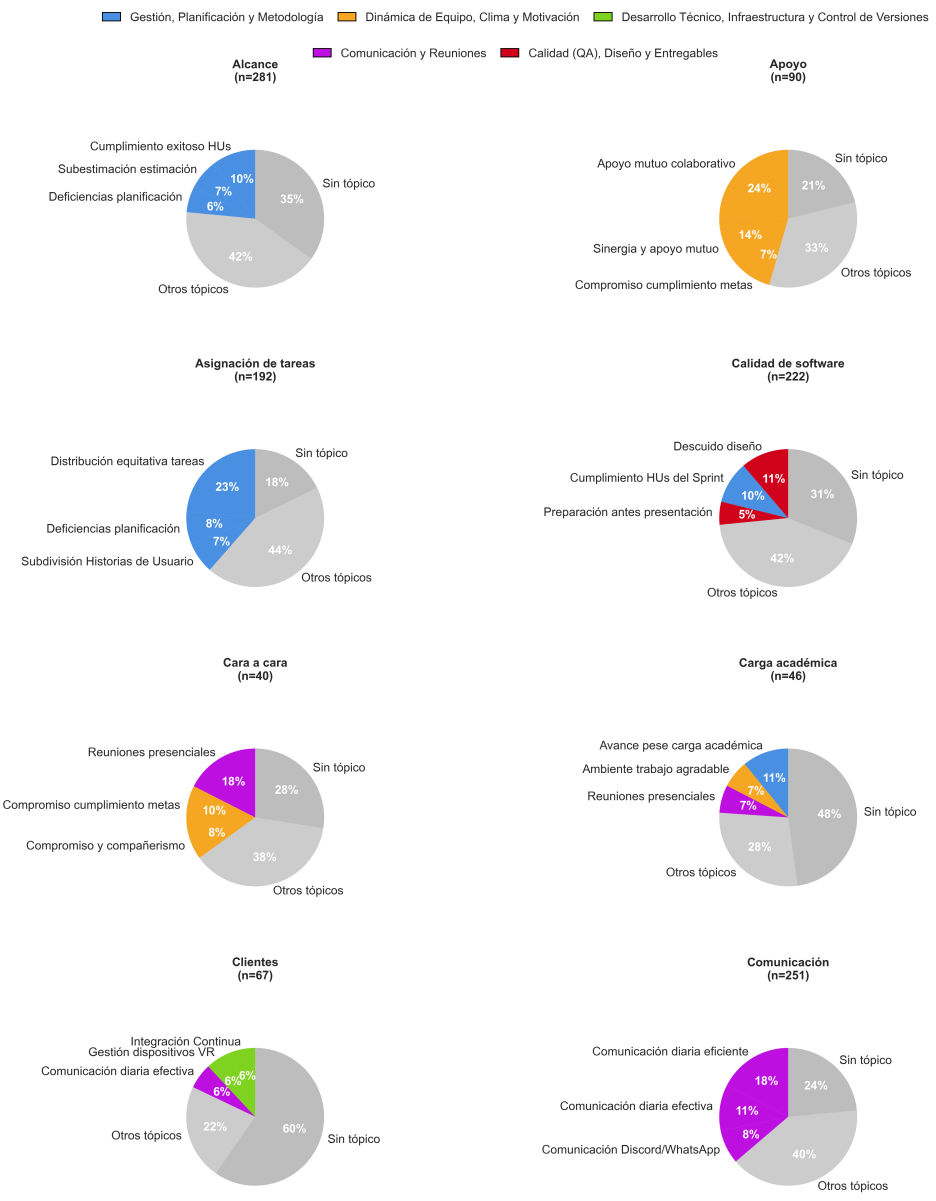


Figura 17: Presencia de Tópicos en los Temas (Parte 1).

proporción de “Sin Tópico” (60 %) sugiere interacciones muy heterogéneas, aunque la aparición de “Integración Continua” y dispositivos específicos indica que el cliente se menciona en función de los entregables técnicos.

Continuando con la Figura 18 (Parte 2), se profundiza en los puntos de dolor técnicos y metodológicos. El contexto de los **Criterios de Aceptación** es mayoritariamente negativo, anclado en “Deficiencias en Criterios” (15 %). El **Diseño** sufre del mismo mal, con un 19 % de las frases apuntando explícitamente al “Descuido de diseño”. Sin embargo, el análisis inductivo valida

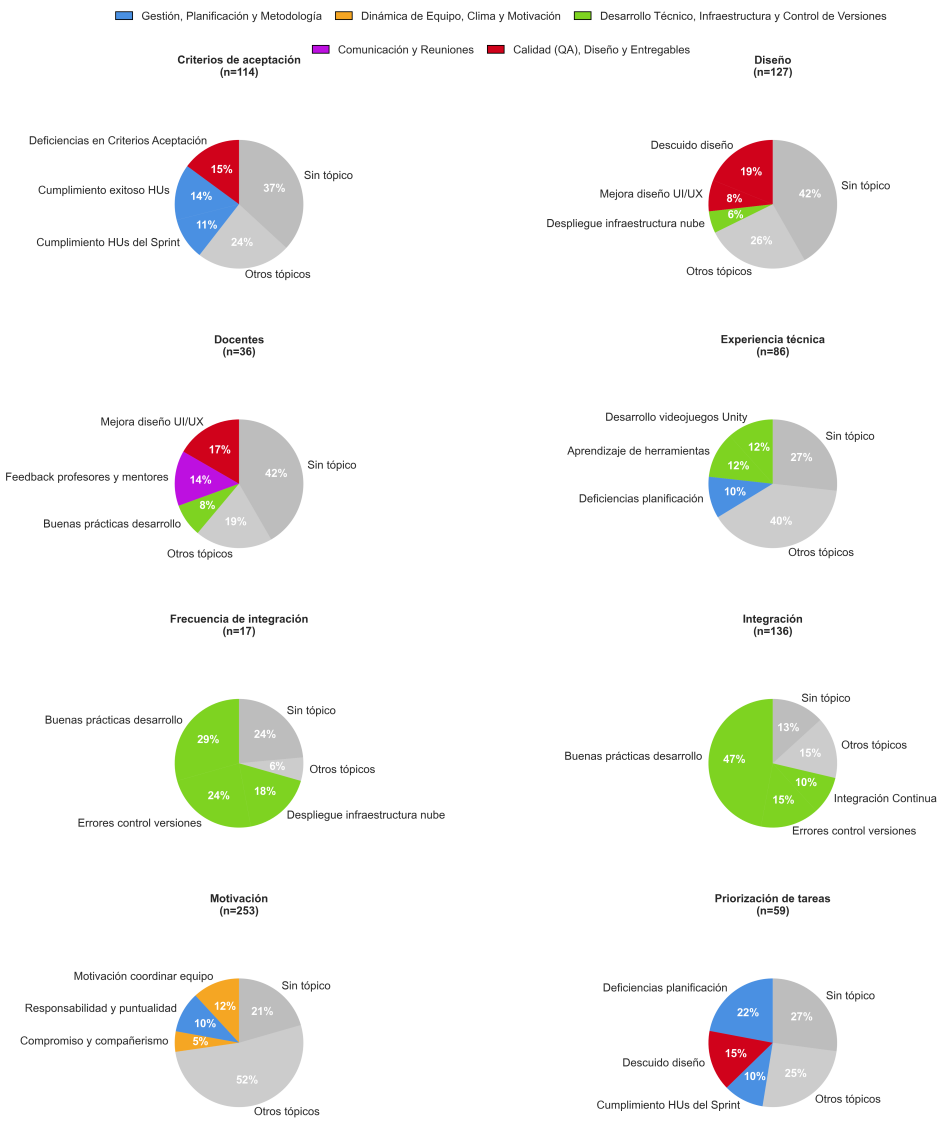


Figura 18: Presencia de Tópicos en los Temas (Parte 2).

la coherencia del etiquetado en Temas técnicos: **Integración** está compuesto en un 47 % por “Buenas prácticas” y un 15 % por “Errores de control de versiones”, aclarando que, para los estudiantes, integrar el software es sinónimo de gestionar Git (la herramiento de control de versiones que usan) correctamente. La **Priorización de tareas** ofrece un hallazgo crítico para el diagnóstico de problemas: su Tópico principal es “Deficiencias de planificación” (22 %), lo que demuestra que la dificultad para priorizar no es un evento aislado, sino una consecuencia directa de una mala estrategia inicial. Por otro lado, la **Experiencia técnica** se contextualiza positivamente en el aprendizaje, mencionando herramientas concretas como Unity dado el contexto particular en donde existe una alta presencia de proyectos que trabajaron en el desarrollo de videojuegos.

Figura 19: Presencia de Tópicos en los Temas (Parte 3).

Finalmente, la Figura 19 (Parte 3) cierra el análisis con los Temas más voluminosos. **Ritmo**, el Tema más frecuente del corpus, revela su naturaleza conflictiva: se compone tanto de “Cumplimiento de HUs” (11 %) como de “Deficiencias de planificación” (10 %) y “Cumplimiento de plazos” (4 %). Esto indica que el ritmo es el termómetro principal del proyecto, discutiéndose en cada Sprint para evaluar si se logró o no la meta. Las **Relaciones** confirman su rol de “amortiguador” social, definidas por “Respeto y confianza” (13 %) y “Ambiente grato” (11 %). Las **Reuniones** se contextualizan casi exclusivamente en la “Reunión diaria (Scrum)” (11 %), validando la adopción de la ceremonia principal de Scrum. Es notable también el pragmatismo del Tema **Tecnologías**, donde el Tópico “Buenas prácticas” (20 %) convive con herramientas de gestión como “Trello y Excel” (12 %), sugiriendo que la tecnología, cuando se discute en las retrospectivas, se valora en función de cómo ordena el caos del proyecto.

La integración de estos hallazgos permite concluir que el contexto de las menciones no es aleatorio. Los Temas de gestión (*Ritmo*, *Alcance*, *Priorización*) aparecen sistemáticamente ligados a Tópicos de deficiencia y subestimación, constituyendo el núcleo problemático del curso. Por el contrario, los Temas sociales (*Apoyo*, *Relaciones*, *Comunicación*) emergen consistentemente en contextos de colaboración y respeto, representando el mayor éxito de la

experiencia *capstone* que se permite obtener del análisis de las retrospectivas.

Este último análisis también sugiere fuertemente que el etiquetado del modelo es coherente con los patrones de información que emergen propiamente de los resúmenes al revisar la coherencia que tienen los Tópicos más presentes en cada uno de los Temas estudiados con la etiqueta y descripción de cada Tema. Puede entenderse entonces que esta triangulación de los resultados inductivos y deductivos le otorga a la metodología (y a los métodos y técnicas utilizados en ella) una validación empírica de la capacidad de representar fielmente el contenido de las retrospectivas.

5.3. Análisis de la Riqueza de la reflexión y su evolución entre Sprints

Para responder a las preguntas de investigación ligadas a la riqueza y su evolución a lo largo de los Sprints (**RQ2** y **RQ3**), es necesario trascender el análisis de frecuencia individual para examinar la diversidad semántica de la discusión. En este contexto, definimos la "Riqueza" de la reflexión como la amplitud de Temas o Tópicos distintos que un equipo es capaz de abordar en una única sesión. Una alta riqueza sugiere una capacidad de escaneo panorámico del estado del proyecto, mientras que una baja riqueza indicaría un enfoque tubular o focalizado en pocos asuntos críticos.

La Figura 20 presenta la distribución de la cantidad de Temas distintos abordados por equipo en cada hito. Al analizar estos datos, se observa un hallazgo notable respecto a la **RQ2**: la reflexión de los equipos es consistentemente variada. La mediana se sitúa en torno a los **11 Temas distintos** por retrospectiva. Considerando que la taxonomía del modelo constó de 21 etiquetas validadas sistemáticamente, esto implica que, en promedio, los equipos discuten cerca del 50 % de todo el espectro temático posible en cada sesión. Esta amplitud evidencia que las retrospectivas no operan como reuniones monotemáticas, sino como instancias integrales donde se revisa el proyecto desde múltiples dimensiones simultáneamente.

Esta visión se complementa con el análisis inductivo presentado en la Figura 21, que mide la riqueza respecto a la cantidad de Tópicos distintos encontrados. Aquí, la mediana oscila entre 8 y 9 Tópicos por sesión, una cifra ligeramente inferior a la de los Temas, lo que indica que aunque se tocan muchas categorías generales, la discusión específica suele aterrizar en un número más acotado de situaciones o "contextos" concretos. Es relevante notar que la dispersión de los datos tiende a disminuir hacia el Sprint 3 en ambas métricas, sugiriendo que, a medida que el proyecto avanza, los equipos convergen hacia un patrón de discusión más estandarizado.

Abordando la **RQ3** sobre la evolución de la reflexión, se analizó la tasa de recambio de contenidos entre Sprints consecutivos. Las Figuras 22 y 23 ilustran un patrón claro de estancamiento temático. En la transición del Sprint 1 al 2, los equipos mantienen una media de 8.73 Temas, incorporando apenas una media de 2.15 Temas nuevos. Esta tendencia a la fijación se agudiza en la transición final hacia el Sprint 3, donde la cantidad media de Temas nuevos

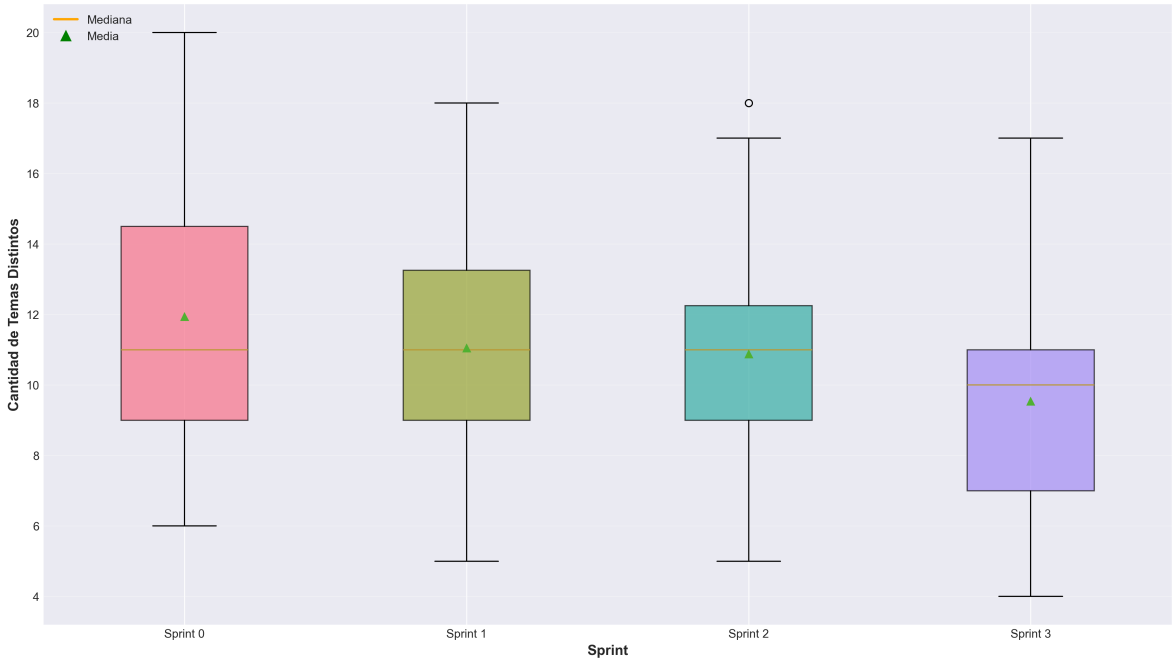


Figura 20: Riqueza de Temas por Sprint.

cae a 1.32, elevando la tasa de mantención al 84.08 %. Esto indica que la inmensa mayoría de la discusión es una repetición recursiva de los asuntos tratados en la sesión anterior.

El análisis granular de los Tópicos (Figuras 24 y 25) agrava este diagnóstico. Al comparar el Sprint 2 con el 3, el 27.66 % de los equipos (13 de 47) repitieron completamente su espectro de Tópicos, sin introducir ninguna novedad.

Sin embargo, afirmar que existe “estancamiento” requiere discernir si lo que se repite son los problemas (inercia negativa) o los éxitos (consolidación positiva). Para dilucidar esto de manera sencilla y utilizando los Temas de interés, la Tabla 10 detalla la cantidad de equipos que persistieron en discutir un mismo Tema en la misma sección (“Qué salió bien” vs. “Qué salió mal”) a través de Sprints consecutivos.

La Tabla 10 ofrece evidencia sobre la naturaleza de la evolución reflexiva. En la columna de persistencia negativa (“Mal”), el Tema **Ritmo** es, por lejos, el más crónico: 28 equipos repitieron quejas sobre su ritmo entre el Sprint 1 y 2, y 21 equipos lo hicieron entre el Sprint 2 y 3. Similarmente, la persistencia en quejas sobre **Calidad de software** aumenta hacia el final del curso (de 18 a 21 equipos), indicando que la deuda técnica no se resuelve, sino que se acumula.

En contraste, la columna de persistencia positiva (“Bien”) muestra dónde reside la fortaleza del curso. Las **Reuniones** son consistentemente celebradas por más de 20 equipos en cada transición. Aún más revelador es el comportamiento de los Temas puramente humanos: **Relaciones** y **Apoyo** tienen *cero* equipos quejándose persistentemente de ellos (columnas

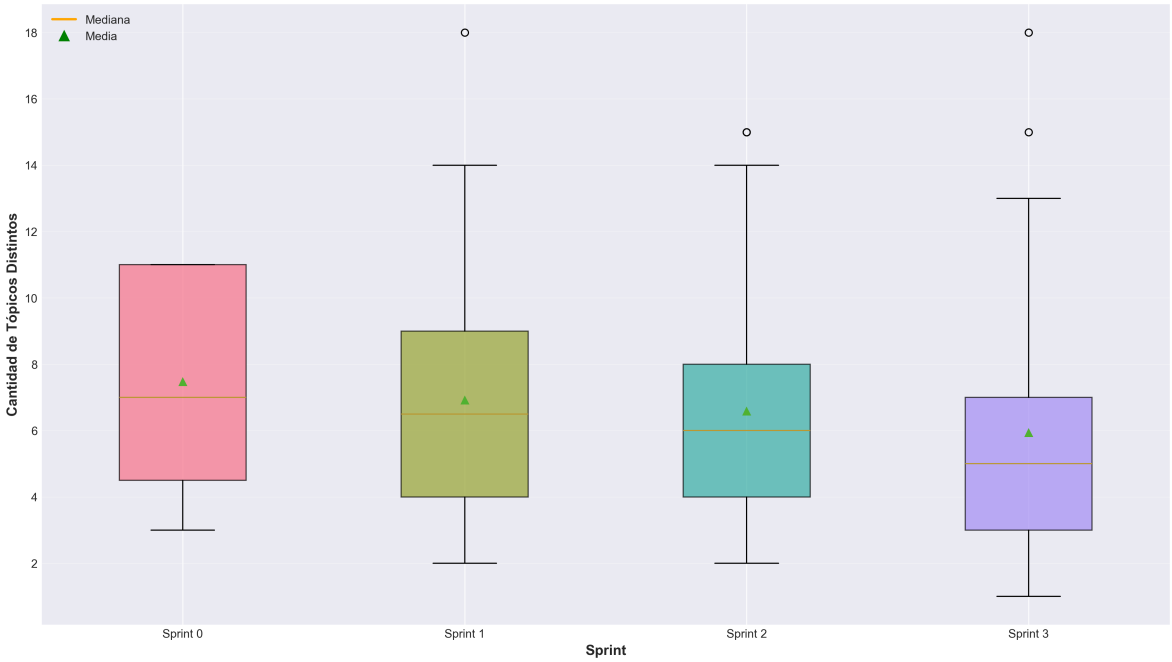


Figura 21: Riqueza de Tópicos por Sprint.

"Mal" en 0), mientras que mantienen una base constante de menciones positivas.

Estos hallazgos permiten guiar la discusión hacia un concepto fundamental para la **RQ3**: la persistencia de los mismos Tópicos y la escasa aparición de nuevos elementos hacia el final del curso no es inocua. La repetición crónica de quejas sobre Ritmo y Calidad sugiere que los problemas de gestión y técnicos detectados anteriormente no se resuelven, sino que se vuelven condiciones basales del trabajo. Así, la retrospectiva cumple eficazmente su función de visibilizar el estado integral del equipo y sostener el vínculo social (como demuestra la persistencia positiva en Reuniones y Relaciones), pero parece tener limitaciones importantes como motor de mejora continua operativa. Los equipos no logran usar la retrospectiva para transformar su realidad operativa, convirtiendo la ceremonia en un espacio de **contención recursiva** —donde se ventilan frustraciones recurrentes y se celebra la mutua convivencia y los éxitos organizacionales— más que en una herramienta de transformación evolutiva ágil que permita transformar el comportamiento de los estudiantes hacia mejorar esos aspectos operativos del proyecto.

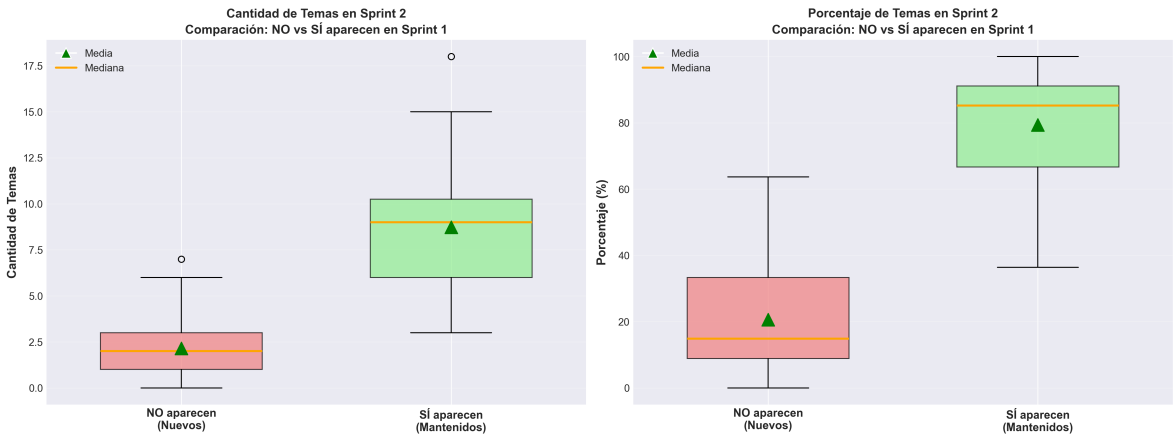


Figura 22: Evolución de Temas: Sprint 2 vs Sprint 1, tanto en cantidad (panel izquierdo) como en porcentaje (panel derecho) de Temas nuevos y mantenidos.

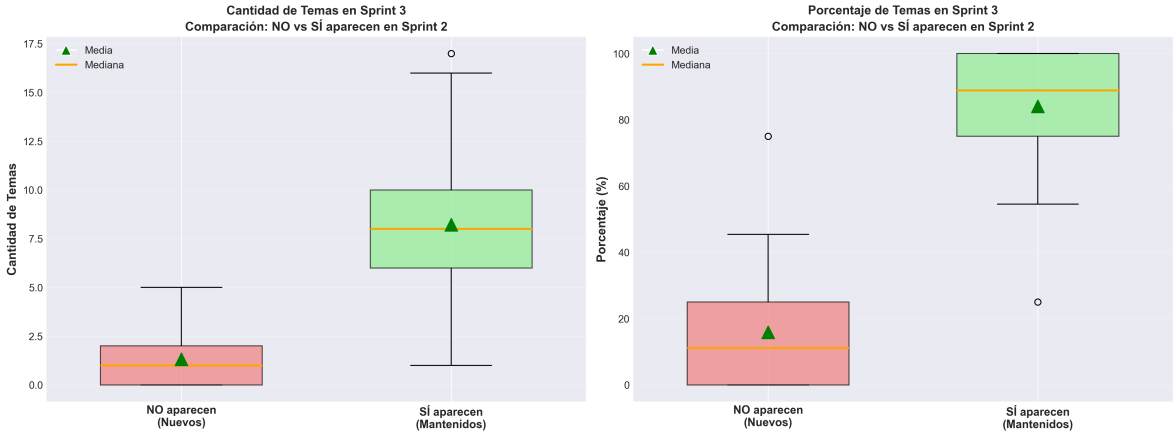


Figura 23: Evolución de Temas: Sprint 3 vs Sprint 2, tanto en cantidad (panel izquierdo) como en porcentaje (panel derecho) de Temas nuevos y mantenidos.

CAPÍTULO 6

CONCLUSIONES

El presente trabajo de tesis abordó el desafío de caracterizar y comprender la práctica reflexiva de los estudiantes de ingeniería en el contexto de proyectos *capstone* de desarrollo de software abordando 3 cohortes de un proyecto realizado en la Universidad Técnica Federico Santa María que se realizaron en el Campus San Joaquín. Buscando explorar y profundizar acerca del proceso de aprendizaje en entornos ágiles, se diseñó, validó e implementó sistemáticamente una metodología de análisis inteligente híbrida que combinó técnicas de Procesamiento de Lenguaje Natural (PLN) deductivas e inductivas. Mediante la clasificación supervisada asistida por Grandes Modelos de Lenguaje (LLMs) y el modelado de Tópicos no

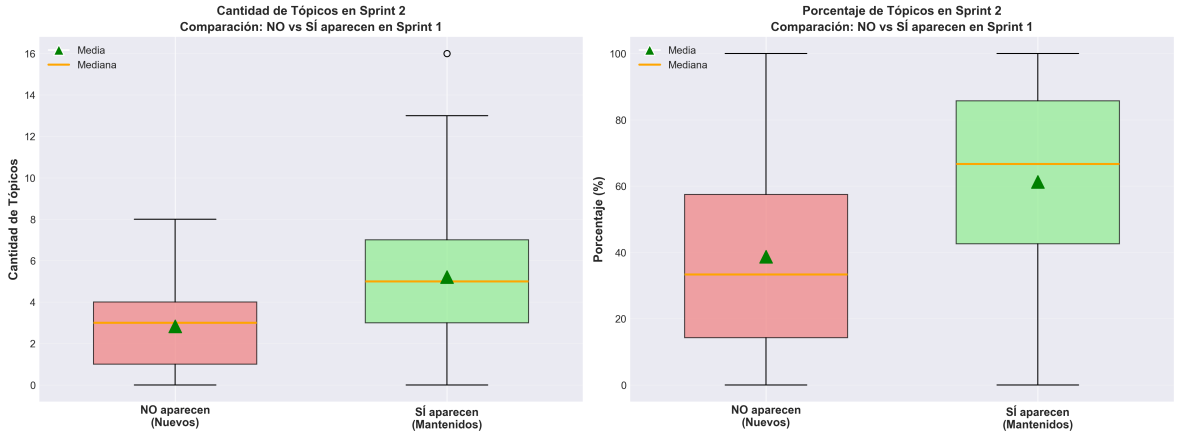


Figura 24: Evolución de Tópicos: Sprint 2 vs Sprint 1, tanto en cantidad (panel izquierdo) como en porcentaje (panel derecho) de Tópicos nuevos y mantenidos.

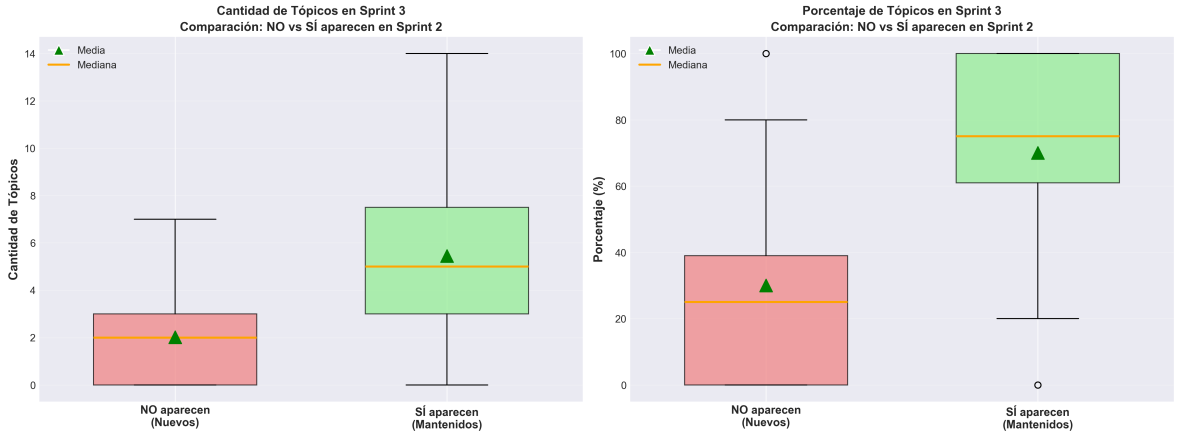


Figura 25: Evolución de Tópicos: Sprint 3 vs Sprint 2, tanto en cantidad (panel izquierdo) como en porcentaje (panel derecho) de Tópicos nuevos y mantenidos.

supervisado (BERTopic), se logró procesar y estructurar un corpus de 2219 frases provenientes de 156 retrospectivas de Sprint a lo largo de tres años académicos (2022-2024).

La metodología propuesta demostró ser robusta, logrando una cobertura del 95.9 % de la información contenida en los resúmenes. Esto permitió transformar texto no estructurado en un diagnóstico cuantitativo y cualitativo sobre cómo los estudiantes perciben, discuten y gestionan sus proyectos. A diferencia de los enfoques tradicionales de análisis manual, limitados en escala y propensos a la subjetividad, esta aproximación permitió identificar patrones transversales de comportamiento, fricciones operativas crónicas y dinámicas sociales resilientes, ofreciendo una radiografía fiel de la experiencia educativa en el curso “Feria de Software”.

El análisis de la presencia de Temas reveló una clara predominancia de la dimensión opera-

Tabla 10: Persistencia de Temas por Sección y transición de Sprints.

Tema	**S1 → S2 (Bien)**	**S1 → S2 (Mal)**	**S2 → S3 (Bien)**	**S2 → S3 (Mal)**
Ritmo	17	**28**	19	**21**
Calidad de software	5	18	7	**21**
Alcance	16	21	19	10
Diseño	4	6	5	7
Tecnologías	13	8	9	5
Reporte de estado	19	7	14	5
Priorización de tareas	0	4	0	3
Comunicación	15	2	17	3
Integración	7	4	9	2
Criterios de aceptación	6	5	8	2
Carga académica	3	1	2	2
Asignación de tareas	13	1	14	2
Experiencia técnica	5	1	5	4
Reuniones	**26**	11	**21**	6
Motivación	17	3	11	1
Clientes	5	1	3	1
Frecuencia de integración	2	0	1	1
Relaciones	7	**0**	6	**0**
Apoyo	10	**0**	9	**0**
Docentes	2	0	2	0
Cara a cara	4	0	3	0

tiva y de gestión por sobre la discusión puramente técnica. Las etiquetas relacionadas con el **Ritmo**, el **Alcance** y las **Reuniones** dominaron el discurso estudiantil, evidenciando que el principal desafío percibido no es la construcción del software *per se*, sino la coordinación del trabajo y la autogestión del equipo. El análisis de sentimientos asociado a estos Temas expuso una dicotomía fundamental: mientras que los aspectos metodológicos (Ritmo, Planificación) son consistentemente negativos y fuentes de frustración, los aspectos humanos (Relaciones, Apoyo, Motivación) actúan como amortiguadores positivos y constantes a lo largo del proyecto.

Por su parte, el análisis inductivo de tópicos validó y profundizó estos hallazgos, proporcionando una granularidad semántica que las etiquetas generales no podían ofrecer. Se descubrió que Temas abstractos como “Integración” se traducen en la práctica en discusiones específicas sobre “Errores de control de versiones”, y que la “Tecnología” se valora principalmente en función de su capacidad para ordenar el caos (herramientas de gestión). La coherencia entre los Tópicos descubiertos por BERTopic y las etiquetas asignadas por el LLM reforzó la validez interna del estudio, confirmando que los estudiantes articulan sus proble-

mas de manera consistente, centrando su narrativa en la dificultad de la planificación y la fortaleza del vínculo grupal.

Finalmente, el análisis de la riqueza y evolución de la reflexión arrojó luz sobre la dinámica temporal de los equipos. Si bien las retrospectivas mostraron una alta riqueza estática —cubriendo una mediana de 11 Temas distintos por sesión—, su evolución temporal reveló un patrón preocupante de estancamiento. A medida que avanzaban los Sprints, la tasa de incorporación de nuevos Temas disminuyó drásticamente y la tasa de mantención de Tópicos aumentó, indicando que los equipos tienden a caer en bucles recursivos de discusión. Lejos de resolver los problemas operativos detectados al inicio, los estudiantes aprenden a convivir con ellos, perdiendo la retrospectiva su capacidad de funcionar como una herramienta efectiva de mejora continua de procesos.

Con estos hallazgos, es posible dar respuesta a cada una de las preguntas de investigación planteadas para el caso estudiado.

6.1. RQ1: ¿Qué Temas, con qué frecuencia y bajo que contexto son aludidos estos en las Retrospectivas de Sprint?

Los resultados indican inequívocamente que las retrospectivas están dominadas por Temas de **Gestión y Operatividad**. Las categorías de *Ritmo*, *Reuniones* y *Alcance* ocupan los primeros lugares en frecuencia, desplazando a Temas técnicos como *Calidad de Software* o *Diseño* a un segundo plano. Esto revela que la carga cognitiva de los estudiantes está puesta mayoritariamente en intentar "domar" el proceso de desarrollo y cumplir con los compromisos adquiridos.

En cuanto al contexto y la connotación, se identificaron tres patrones claros:

1. **Contexto de Fricción:** Los Temas de gestión (Ritmo, Alcance, Asignación) se aluden predominantemente en contextos negativos, asociados a la sección "¿Qué salió mal?". Se discuten como obstáculos persistentes derivados de la subestimación del esfuerzo y la falta de disciplina en la planificación.
2. **Contexto de Soporte:** Los Temas sociales (Relaciones, Apoyo, Motivación) aparecen en contextos altamente positivos y de celebración ("¿Qué salió bien?"). Se aluden como el factor que permite al equipo seguir adelante a pesar de las dificultades técnicas u organizativas.
3. **Contexto Instrumental:** La tecnología y las reuniones se discuten con una connotación mixta pero pragmática, valorándose en la medida en que habilitan la coordinación y el flujo de trabajo.
4. **Contexto de Calidad:** Los Temas ligados a las funcionalidades desarrolladas de los sistemas de software como "Calidad de Software", "Criterios de Aceptación" y "Diseño"

fueron mencionados mayoritariamente con connotación negativa, ya sea mostrando que el nivel de calidad o que el comportamiento deseado no fue el logrado o que fue uno de los aspectos deficientes de cada Sprint, mantuviendose constante incluso en Sprints tardíos.

Respecto a las implicancias que esto puede tener para los instructores y el diseño del curso, esto sugiere que la intervención docente no debe centrarse tanto en enseñar nuevas tecnologías, sino en proporcionar andamiajes más fuertes para la gestión de proyectos y en técnicas ligadas al uso de herramientas y tecnologías para mejorar el manejo de la deuda técnica de los estudiantes. La alta negatividad en Ritmo y Alcance es una señal de alerta de que los estudiantes carecen de herramientas efectivas para estimar, planificar y progresar consistentemente en escenarios de incertidumbre, una competencia que debería reforzarse antes o durante el curso capstone.

6.2. RQ2: ¿Qué tan variada o rica es la reflexión de los equipos cuando realizan Retrospectivas de Sprint?

La reflexión de los equipos es **sorprendentemente rica y variada** en su corte transversal. Con una mediana consistente de 11 Temas distintos y entre 8 y 9 Tópicos específicos abordados por sesión, los estudiantes demuestran capacidad para realizar un escaneo panorámico del estado de su proyecto, aunque no necesariamente abordando todos los Temas de interés que se estudiaron. De todas formas, los estudiantes no se limitan a discutir un solo dolor urgente, sino que logran articular una visión que integra simultáneamente dimensiones técnicas (código, integración), metodológicas (scrum, ritmo) y humanas (relaciones, motivación).

Este hallazgo valida el formato de la retrospectiva como un instrumento pedagógico efectivo para fomentar la visión holística de la ingeniería de software. Los estudiantes, incluso siendo novatos, son capaces de identificar y reconocer a través de la síntesis de los aspectos más relevantes de la discusión, la naturaleza multidimensional de sus proyectos.

6.3. RQ3: ¿Cómo evoluciona la reflexión de los equipos entre Sprints cuando realizan Retrospectivas de Sprint?

La reflexión evoluciona desde una fase inicial de exploración y ajuste hacia una fase de **fijación y estancamiento**. Si bien en los primeros Sprints hay una incorporación saludable de Temas nuevos, hacia la segunda mitad del proyecto la discusión se vuelve rígida. Se observó que, en la transición del Sprint 2 al 3, la tasa de mantención de Temas supera el 84 %, y más de un cuarto de los equipos repite exactamente el mismo espectro de Tópicos sin introducir ninguna novedad, lo que es aún más preocupante.

Esta evolución sugiere que los equipos entran en un estado de inercia donde la reflexión deja de ser efectiva como una herramienta de mejora continua, dejando de ser un motor de adaptación ágil (donde se prueba una solución, se evalúa y se cambia) para convertirse en un registro recursivo de las mismas dificultades operativas.

Este comportamiento alerta sobre la necesidad de intervenciones docentes a mitad del semestre. Los instructores deben estar atentos a la repetición de Tópicos en las bitácoras o resúmenes. Si un equipo reporta "problemas de ritmo" en dos Sprints consecutivos, es indicativo de que carecen de la autonomía para resolverlo por sí mismos o que están renunciando a usar la retrospectiva como una fuente valiosa de acción para el cambio en sus procesos. En este sentido, sugerencias para el diseño del curso pueden ir en la línea de implementar hitos de control que fuercen un cambio de estrategia o la introducción de restricciones externas que rompan la inercia del equipo.

6.4. RQ4: ¿Cuáles son los problemas y éxitos más importantes que se pueden extraer del análisis y en qué contexto aparecen dentro de las Retrospectivas de Sprint?

Como diagnóstico medular de la experiencia *capstone*, esta pregunta busca identificar las dimensiones estructurales donde los estudiantes experimentan sus mayores victorias y sus fracasos más persistentes. Al integrar los hallazgos de frecuencia, sentimiento y evolución, se configura un perfil claro del comportamiento de los equipos que revela una tensión constante entre la capacidad humana y la disciplina metodológica.

El problema más crítico y transversal que enfrentan los estudiantes es la **Fricción Operativa Crónica** derivada de una deuda de gestión. La evidencia es contundente al respecto: el Tema "Ritmo" no solo es el más frecuente del corpus, sino que en su mayoría mantiene una carga negativa constante y dominante desde el inicio hasta el final del proyecto. Los estudiantes discuten sistemáticamente este aspecto en la sección "¿Qué salió mal?", contextualizándolo predominantemente bajo Tópicos específicos de "Deficiencias de planificación" y "Subestimación de la estimación". La persistencia de estas quejas, que se repiten en 28 equipos entre el Sprint 1 y 2, demuestra que los estudiantes son capaces de identificar su incapacidad para planificar y mantener un ritmo sostenible, pero carecen de las herramientas o la experiencia para resolver este problema por sí mismos, convirtiéndolo en una condición crónica que sufren durante todo el semestre.

En contraposición a esta dificultad operativa, los estudiantes demuestran una notable capacidad para construir **Resiliencia Social**, siendo este su éxito más significativo. El logro basal de los equipos es la consolidación de un capital humano robusto, evidenciado por la polaridad abrumadoramente positiva de Temas como "Relaciones", "Apoyo" y "Motivación". Estos aspectos dominan la sección "¿Qué salió bien?" y se discuten en el contexto de la colaboración efectiva, validado por la alta presencia del Tópico "Apoyo mutuo colaborativo". A diferencia

de los problemas técnicos o de gestión, las dinámicas relacionales no se deterioran con la presión del cierre del proyecto; por el contrario, actúan como el amortiguador psicológico que permite al equipo sobrevivir a la frustración del desarrollo sin desintegrarse.

En una dimensión técnica, se identifica un problema emergente relacionado con la **Visibilidad Tardía de la Deuda Técnica**. Existe una desconexión temporal entre la ejecución y la calidad, donde los estudiantes tienden a priorizar la entrega de funcionalidades (Ritmo) por sobre la calidad interna. Esto genera una "bomba de tiempo" que explota hacia el final del curso, como lo demuestra el incremento en la frecuencia y negatividad de Temas como "Calidad de Software" y "Diseño" en los últimos Sprints. La asociación de estos Temas con Tópicos de "Descuido de diseño" y "Errores de control de versiones" indica que los equipos tienen dificultades para integrar prácticas de calidad preventiva, enfrentándose a las consecuencias de sus decisiones técnicas solo cuando la complejidad del sistema escala y la refactorización se vuelve costosa.

Por otro lado, se observa un éxito instrumental en la **Valoración de la Sincronización**. Contrario a la percepción común de que las ceremonias ágiles pueden verse como burocracia innecesaria, los estudiantes validan y defienden la utilidad de los rituales de coordinación. El Tema "Reuniones" es el segundo más frecuente y tiene una alta presencia en la sección "¿Qué mantener?" con sentimiento positivo. El contexto de esta valoración es pragmático: el Tópico predominante es "Reuniones diarias (Scrum)", lo que indica que el reunirse diariamente ha sido adoptado y reconocido como una buena práctica y como el mecanismo principal para reducir la incertidumbre y coordinar el trabajo diario, funcionando efectivamente como el instrumento operativo de cohesión del equipo.

Finalmente, existe un problema de orden meta-cognitivo que se puede denominar como **Estancamiento Reflexivo**. A pesar de realizar la ceremonia de retrospectiva regularmente, los equipos pierden capacidad de adaptación a medida que avanza el curso. La drástica caída en la aparición de nuevos Temas y la repetición exacta de los mismos Tópicos entre el Sprint 2 y 3 revelan que los estudiantes normalizan sus disfunciones y que simplemente repiten los Temas discutidos en iteraciones anteriores. Esto es preocupante, ya que implica que la retrospectiva deja de ser un motor de transformación ágil para convertirse en un espacio de contención recursiva o simplemente su valor para ser el motor de transformación es desechado por los estudiantes, donde se ventilan las mismas frustraciones o éxitos, lo cual representa una desviación crítica del propósito de la mejora continua.

El diagnóstico general es que los estudiantes son **resilientes socialmente pero frágiles metodológicamente**. El curso logra exitosamente fomentar el trabajo en equipo y la adopción de rituales básicos de sincronización, pero evidencia brechas críticas en la formación de competencias de autogestión y disciplina técnica preventiva. Para los instructores, esto sugiere la necesidad de rebalancear el apoyo docente: intervenir tempranamente en la planificación para evitar la deuda de gestión, monitorear la calidad del código en hitos intermedios para visibilizar la deuda técnica antes del final, y actuar activamente para romper la inercia cuando los equipos caen en la repetición de sus propios problemas, forzando un cambio de estrategia que los estudiantes no están logrando generar por sí mismos.

6.5. Trabajo Futuro

A partir de los hallazgos presentados y las limitaciones naturales de este estudio, emergen diversas líneas de investigación que permitirían profundizar en la comprensión y el apoyo a la práctica reflexiva en la formación de ingenieros de software.

Una primera línea de investigación natural es el cruce de los patrones reflexivos detectados con métricas objetivas de desempeño. Si bien esta tesis ha caracterizado la percepción subjetiva de los estudiantes sobre su proceso (sentimientos, fricciones, éxitos), sería valioso correlacionar estos datos con indicadores cuantitativos de éxito académico y calidad del producto. Investigaciones futuras podrían analizar si los equipos que mantienen una discusión rica y variada obtienen mejores calificaciones finales, o si aquellos que reportan crónicamente problemas de "Calidad de Software" presentan efectivamente una mayor deuda técnica o una menor cobertura de pruebas en sus repositorios de código. Este análisis permitiría validar si la madurez reflexiva es un predictor confiable de la calidad ingenieril.

En segundo lugar, se propone expandir el alcance del análisis hacia la componente proactiva de las retrospectivas. El presente estudio se centró en el diagnóstico que hacían los estudiantes de su situación (a través de las secciones qué salió bien y qué salió mal), pero una retrospectiva completa debe desembocar en un plan de acción. Sería relevante aplicar la misma metodología híbrida para analizar la sección de "Acciones a tomar" o "Compromisos" y como estas impactan en el rendimiento académico de los equipos. Esto permitiría investigar si la "fricción operativa crónica" y el "estancamiento reflexivo" detectados se deben a que los equipos no proponen soluciones, o a que proponen soluciones vagas e ineficaces que no logran implementar, dando luces sobre la calidad de los planes de mejora.

Desde una perspectiva tecnológica y aplicada, la validación de la metodología de detección automática de Temas abre la puerta al desarrollo de herramientas de intervención en tiempo real. Dado que es factible identificar patrones de estancamiento o focos de negatividad mediante algoritmos, un trabajo futuro lógico es la construcción de *dashboards* o asistentes virtuales que procesen las retrospectivas al momento de su escritura. Estas herramientas podrían alertar a los docentes cuando un equipo cae en la repetición de un problema crítico (como el Ritmo) durante dos o más Sprints, o sugerir a los estudiantes estrategias de mitigación específicas basadas en la naturaleza de su discusión, transformando el análisis *post-mortem* en una asistencia formativa activa.

Por otro lado, y bajo el mismo espíritu de esta tesis, también resulta interesante continuar investigando si existen maneras automáticas e inteligentes de medir la "profundidad" de la reflexión, es decir, si los estudiantes discuten de manera superficial o profunda los aspectos a los que se refieren en las retrospectivas de Sprint. Esto permitiría entender con mayor claridad si la discusión que realizan los estudiantes permite el aprendizaje de los aspectos más complejos de este tipo de proyectos, respaldando así que la reflexión es también un medio para sentar ideas e interiorizar con mayor fuerza la práctica profesional que se busca enseñar en los proyectos capstone.

Finalmente, sería enriquecedor realizar estudios comparativos que contrasten la realidad estudiantil con la práctica industrial. Replicar esta metodología en un corpus de retrospectivas de equipos profesionales *junior* y *senior* permitiría dimensionar la brecha real entre la academia y la industria. Esto ayudaría a determinar cuánto de la "fragilidad metodológica" observada es propia del proceso de aprendizaje universitario y cuánto es inherente a la naturaleza del desarrollo de software ágil, permitiendo ajustar los currículos para enfocar la enseñanza en aquellas competencias de autogestión que resulten ser las más críticas para el desempeño profesional.

Anexos

7.1. Información de Tópicos

La Tabla 11 presenta el detalle de los 69 tópicos identificados, excluyendo la descripción larga para facilitar la lectura.

Tabla 11: Información de tópicos: ID, descripción corta, palabras clave principales y cantidad de frases.

ID	Descripción Corta	Palabras Clave	Frases
0	Pair Programming para tareas	pantalla, programación, sesiones, pair, programming	13
1	Gestión dispositivos VR	funcionaba, vr, realidad, virtual, oculus	25
2	Responsabilidad y puntualidad	entregas, las, reuniones, responsabilidad, puntualidad	30
3	Despliegue infraestructura nube	servidor, backend, frontend, datos, google	33
4	Subequipos especializados	funcionando, backend, frontend, o, js	12
5	Uso Jira y Toggl	uso, tiempos, toggl, registro, jira	24
6	Buenas prácticas desarrollo	idioma, adoptó, buenas, prácticas, código	10
7	Errores control versiones	versiones, repositorio, mal, grandes, commits	34
8	Integración Continua	integración, entorno, continua, fácilmente, accesible	20
9	Buenas prácticas desarrollo	merge, utilización, git, github, ramas	74
10	Desarrollo IA y LLMs	estudiantes, prompts, inteligencia, artificial, ia	18
11	Desarrollo videojuegos Unity	videojuegos, investigar, tecnológicas, implementados, unity	17
12	Investigación tecnologías	nuevas, apoyen, estudio, implementaciones, tecnologías	9
13	Comunicación Discord/WhatsApp	medios, coordinación, whatsapp, mediante, discord	27
14	Seguimiento horas HUs	motiva, método, dedicadas, horas, hu	9

ID	Descripción Corta	Palabras Clave	Frases
15	Uso Trello y Excel	excel, tareas, uso, organizar, trello	33
16	Feedback profesores y mentores	esfuerzo, mejorar, profesores, constructivo, feedback	25
17	Pruebas (testing) y validación	da, funcionalidades, testeadas, corroborar, importancia	13
18	Aprendizaje de herramientas	trabajábamos, aprender, técnicas, utilizadas, herramientas	11
19	Participación equitativa	miembros, consideran, participación, todos, continuo	11
20	Compromiso y compañerismo	desde, exigir, promovió, compañerismo, compromiso	16
21	Motivación e interés equipo	concluir, mantener, desempeño, interés, motivación	16
22	Motivación coordinar equipo	gracias, hemos, ello, motivación, logrado	33
23	Subdivisión Historias de Usuario	subdividiend, historia, pequeñas, subequipos, historias	16
24	Roles definidos y separación	productividad, mantener, marcados, separación, roles	15
25	Compromiso cumplimiento metas	trabajó, integrante, aquellos, asignadas, logrando	28
26	Compromiso y perseverancia	proyecto, perseverar, entregables, sacar, compromiso	10
27	Alta motivación y compromiso	todos, estamos, entregar, motivados, motivación	12
28	Compromiso y compañerismo	compartir, aumentaron, implicancia, juntas, aumentó	16
29	Comunicación diaria efectiva	comunicarse, conversación, mantener, buena, comunicación	31
30	Reuniones sociales y lúdicas	mantener, crucial, ánimo, lúdico, social	9
31	Buena coordinación	hay, sinergia, hubo, buena, coordinación	16
32	Subequipos especializados	subgrupos, apis, separó, end, equipos	11
33	Distribución equitativa tareas	integrante, tareas, habilidades, cada, división	45
34	Comunicación diaria eficiente	cosas, buena, resolver, entre, comunicación	52

ID	Descripción Corta	Palabras Clave	Frases
35	Buena comunicación	conversando, comunicación, disposición, general, buena	8
36	Respeto y confianza equipo	siempre, integrantes, entre, confianza, respeto	22
37	Compañerismo ambiente grato	algo, buena, ambiente, mantener, onda	28
38	Buena onda y cordialidad	simpatía, cordialidad, primera, asignatura, conocernos	10
39	Ambiente trabajo agradable	impulsó, motivó, agradable, empatía, ambiente	19
40	Apoyo mutuo colaborativo	entre, cuando, mutuo, confianza, apoyo	35
41	Responsabilidad en reuniones	constantes, reuniones, 100 %, asistir, asistencia	9
42	Organización y responsabilidad	jerárquica, asumir, descansar, burnout, responsabilidad	11
43	Avance pese carga académica	parte, imágenes, gran, avanzar, logró	20
44	Reuniones presenciales	avanzar, universidad, Sprint, juntas, casa	20
45	Trabajo equipo disposición	actitud, alguien, ayudarlo, necesitaba, disposición	13
46	Sinergia y apoyo mutuo	apoyo, sinergia, sus, obstáculos, cuando	15
47	Mejora diseño UI/UX	consultas, mejorar, ux, ayudante, diseño	11
48	Descuido diseño	aspecto, funcionales, diseño, aplicación, interfaz	31
49	Deuda Técnica y refactorización	medidas, documentando, abordar, deuda, técnica	10
50	Ineficiencia en reuniones	stand, largas, clases, reuniones, ocasiones	26
51	Cumplimiento plazos	con, tareas, resultado, plazos, tiempo	24
52	Responsabilidad en reuniones	poder, está, daily, saber, dailys	36
53	Ineficiencia en reuniones	larga, cortas, ruta, semana, duración	14
54	Reuniones diarias (Scrum)	meetings, mantener, daily, diarias, reuniones	37
55	Incumplimiento deadlines internas	solución, fechas, deadlines, fecha, internas	13

ID	Descripción Corta	Palabras Clave	Frases
56	Efectividad reuniones recurrentes	tanto, faltaba, estuviera, facilitaron, reuniones	17
57	Reuniones diarias (Dailies)	manteniéndonos, costará, informados, reuniones, diarias	15
58	Deficiencias en Criterios Aceptación	descritos, algunos, criterio, criterios, aceptación	18
59	Subestimación estimación	puntos, historia, subestimó, historias, usuario	24
61	Cumplimiento HUs del Sprint	todo, revise, usuarios, usuario, historias	19
62	Deficiencias planificación	que, falta, algunas, mala, no	98
63	Cumplimiento HUs del Sprint	lograron, Sprint, usuario, todas, historias	11
64	Preparación antes presentación	días, antes, día, último, presentación	26
65	Preparación antes presentación	probar, presentación, sistema, tarde, errores	18
66	Cumplimiento HUs del Sprint	casos, no, lo, durante, Sprint	72
67	Dirección Scrum Master y PO	scrum, product, owner, master, avance	10
68	Cumplimiento exitoso HUs	realizaron, propuestas, logró, todas, Sprint	31
69	Buena coordinación	1, en, este, bien, Sprint	48

REFERENCIAS BIBLIOGRÁFICAS

[IEE, 1991] (1991). Ieee standard computer dictionary: A compilation of ieee standard computer glossaries. *IEEE Std 610*, pp. 1–217.

[Adenowo y Adenowo, 2013] Adenowo, A. A. y Adenowo, B. A. (2013). Software engineering methodologies: a review of the waterfall model and object-oriented approach. *International Journal of Scientific & Engineering Research*, 4(7):427–434.

[Aitken e Ilango, 2013] Aitken, A. e Ilango, V. (2013). A comparative analysis of traditional software engineering and agile software development. En *2013 46th Hawaii International Conference on System Sciences*, pp. 4751–4760. IEEE.

[Andriyani *et al.*, 2017] Andriyani, Y., Hoda, R., y Amor, R. (2017). Reflection in agile retrospectives. En *Agile Processes in Software Engineering and Extreme Programming: 18th International Conference, XP 2017, Cologne, Germany, May 22-26, 2017, Proceedings 18*, pp. 3–19. Springer International Publishing.

[Bastarrica *et al.*, 2017] Bastarrica, M. C., Perovich, D., y Samary, M. M. (2017). What can students get from a software engineering capstone course? En *2017 IEEE/ACM 39th International Conference on software engineering: software engineering Education and Training Track (ICSE-SEET)*, pp. 137–145. IEEE.

[Beck *et al.*, 2001] Beck, Kent and Beedle, Mike and Van Bennekum, Arie and Cockburn, Alistair and Cunningham, Ward and Fowler, Martin and Grenning, James and Highsmith, Jim and Hunt, Andrew and Jeffries, Ron and others (2001). The agile manifesto.

[Bi *et al.*, 2025] Bi, Z., Chen, K., Tseng, C.-Y., Zhang, D., Wang, T., Luo, H., Chen, L., Huang, J., Guan, J., Hao, J., y Song, J. (2025). Is GPT-OSS good? a comprehensive evaluation of OpenAI's latest open source models. *arXiv preprint arXiv:2508.12461*. Evaluación independiente que resalta que la variante 20B supera a la 120B en benchmarks como HumanEval y MMLU, ofreciendo una eficiencia superior.

[Bouma, 2009] Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. En *Proceedings of GSCL*, volumen 30, pp. 31–40.

[Brown *et al.*, 2020] Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and Kaplan, Jared D and Dhariwal, Prafulla and Neelakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

[Calefato *et al.*, 2018] Calefato, F., Lanubile, F., Maiorano, F., y Novielli, N. (2018). Sentiment analysis of developers in communication channels: an empirical study. *Empirical Software Engineering*, 23:3752–3789.

[Caliński y Harabasz, 1974] Caliński, T. y Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

[Castellanos *et al.*, 2025] Castellanos, A., Jiang, H., Gomes, P., Vander Meer, D., y Castillo, A. (2025). Large language models for thematic summarization in qualitative health care research: Comparative analysis of model and human performance. *JMIR AI*, 4:e64447.

[Clear y Veling, 2012] Clear, T. y Veling, G. (2012). The design of software engineering capstone projects. En *Proceedings of the 15th international conference on Network-Based Information Systems*, pp. 577–582. IEEE.

[Dieng *et al.*, 2020] Dieng, A. B., Ruiz, F. J., y Blei, D. M. (2020). Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

[Dingsøyr *et al.*, 2018] Dingsøyr, T., Mikalsen, M., Solem, A., y Vestues, K. (2018). Learning in the large-an exploratory study of retrospectives in large-scale agile development. En *Agile Processes in Software Engineering and Extreme Programming: 19th International Conference, XP 2018*, pp. 191–198. Springer.

[Gestwicki y McNely, 2013] Gestwicki, P. V. y McNely, B. J. (2013). Empirical evaluation of periodic retrospective assessment. En *Proceeding of the 44th ACM technical symposium on Computer science education*, pp. 699–704.

[Grootendorst, 2022] Grootendorst, M. (2022). Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

[Herhausen *et al.*, 2025] Herhausen, D., Ludwig, S., Abedin, E., Haque, N. U., y de Jong, D. (2025). From words to insights: Text analysis in business research. *Journal of Business Research*, 198:115491.

[Hoda *et al.*, 2018] Hoda, R., Salleh, N., y Grundy, J. (2018). The rise and evolution of agile software development. *IEEE Software*, 35(5):58–63.

[Hou *et al.*, 2023] Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., Luo, X., Lo, D., Grundy, J., y Wang, H. (2023). Large language models for software engineering: A systematic literature review. *arXiv preprint arXiv:2308.10620*.

[Hundhausen *et al.*, 2024] Hundhausen, C., Conrad, P., Tariq, A., Pugal, S., y Flores, B. Z. (2024). An empirical study of the content and quality of sprint retrospectives in undergraduate team software projects. En *Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training*, pp. 104–114.

[Kato y Van Greunen, 2023] Kato, I. y Van Greunen, D. (2023). Capstone projects and their transition into the software development industry: A 10 year systematic review of literature. En *International Conference on Artificial Intelligence and its Applications*, pp. 114–119.

[Khodeir y Elghannam, 2025] Khodeir, N. y Elghannam, F. (2025). Efficient topic identification for urgent mooc forum posts using bertopic and traditional topic modeling techniques. *Education and Information Technologies*, 30(5):5501–5527.

[Kifetew *et al.*, 2018] Kifetew, F. M., Desta, B. A., Gómez, O. M., y Berntsen, M. (2018). Common pitfalls in software engineering capstone projects: an industrial perspective. En *Proceedings of the 40th International Conference on Software Engineering: Software Engineering Education and Training*, pp. 116–125.

[Kneser y Ney, 1995] Kneser, R. y Ney, H. (1995). Improved backing-off for m-gram language modeling. En *1995 international conference on acoustics, speech, and signal processing*, volumen 1, pp. 181–184. IEEE.

[Kul *et al.*, 2018] Kul, G., Luong, D., Xie, T., Chandola, V., Kennedy, O., y Upadhyaya, S. (2018). Similarity metrics for sql query clustering. *IEEE Transactions on Knowledge and Data Engineering*, 30(12):2408–2420.

[Kurtanović y Maalej, 2017] Kurtanović, Z. y Maalej, W. (2017). Automatically classifying functional and non-functional requirements using supervised machine learning. En *2017 IEEE 25th International Requirements Engineering Conference (RE)*, pp. 490–495. IEEE.

[Lehtinen *et al.*, 2017] Lehtinen, T. O., Itkonen, J., y Lassenius, C. (2017). Recurring opinions or productive improvements—what agile teams actually discuss in retrospectives. *Empirical Software Engineering*, 22:2409–2452.

[Li *et al.*, 2023] Li, Z. S., Arony, N. N., Devathasan, K., y Damian, D. (2023). "software is the easy part of software engineering"-lessons and experiences from a large-scale, multi-team capstone course. En *2023 IEEE/ACM 45th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*, pp. 223–234. IEEE.

[Lutz y Paretti, 2017] Lutz, B. y Paretti, M. C. (2017). Exploring student perceptions of capstone design outcomes. *International Journal of Engineering Education*.

[Ma *et al.*, 2025] Ma, L., Chen, R., Ge, W., Rogers, P., Lyn-Cook, B., Hong, H., Tong, W., Wu, N., y Zou, W. (2025). Ai-powered topic modeling: comparing lda and bertopic in analyzing opioid-related cardiovascular risks in women. *Experimental Biology and Medicine*, 250:10389.

[Maalej y Nabil, 2015] Maalej, W. y Nabil, H. (2015). Bug report, feature request, or simply praise? on automatically classifying app reviews. En *2015 IEEE 23rd international requirements engineering conference (RE)*, pp. 116–125. IEEE.

[Mahnic, 2012] Mahnic, V. (2012). Using scrum framework in a software engineering capstone project. *Electronics and Electrical Engineering*, 18(5):67–70.

[Majanoja y Vasankari, 2018] Majanoja, A.-M. y Vasankari, T. (2018). Reflections on teaching software engineering capstone course. En *CSEDU (2)*, pp. 68–77.

[Marshburn y Sieck, 2019] Marshburn, D. y Sieck, J. (2019). Dont break the build: developing a scrum retrospective game. *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

[Matthies, 2020] Matthies, C. (2020). Playing with your project data in scrum retrospectives. En *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings*, pp. 113–115.

[Matthies y Dobrigkeit, 2019] Matthies, C. y Dobrigkeit, F. (2019). Towards empirically validated remedies for scrum retrospective headaches. *arXiv preprint arXiv:1910.08763*.

[Matthies *et al.*, 2019] Matthies, C., Teusner, R., y Plattner, H. (2019). Ketamine: A tool for investigating the usage of linguistic patterns in sprint retrospectives. En *2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, pp. 55–58. IEEE.

[Ng *et al.*, 2020] Ng, Y. Y., Skrodzki, J., y Wawryk, M. (2020). Playing the sprint retrospective: a replication study. En *Advances in Agile and User-Centred Software Engineering: Third International Conference on Lean and Agile Software Development, LASD 2019, and 7th Conference on Multimedia, Interaction, Design and Innovation, MIDI 2019, Leipzig, Germany, September 1–4, 2019, Revised Selected Papers 3*, pp. 133–141. Springer.

[Nikolenko, 2016] Nikolenko, S. (2016). Topic quality metrics based on distributed word representations. En *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 1029–1032.

[Nikolenko *et al.*, 2015] Nikolenko, S., Koltcov, S., y Koltsova, O. (2015). Topic modelling for qualitative studies. *Journal of Information Science*, 43(1):88–102.

[Östman y Hallmén, 2023] Östman, N. y Hallmén, S. (2023). Exploring development team dialogues through game-based sprint retrospectives.

[Ozoliņš, 2018] Ozoliņš, P. (2018). Preparation and facilitation of retrospective meeting in scrum process. *Information Technology and Management Science*, 21:60–63.

[Palopak y Huang, 2024] Palopak, Y. y Huang, S. J. (2024). Perceived impact of agile principles: Insights from a survey-based study on agile software development project success. *Information and Software Technology*, 176:107552.

[Pereira *et al.*, 2025] Pereira, A., Viegas, F., Dias, D. R. C., Tuler, E., Machado, A. C., Fonseca, G., Gonçalves, M. A., y Rocha, L. (2025). “are the current topic modeling evaluation metrics enough?” mitigating the limitations of topic modeling evaluation metrics using a multi-perspective game theoretic approach. *Knowledge-Based Systems*, p. 113634.

[Przybyłek *et al.*, 2022] Przybyłek, A., Albecka, M., Springer, O., y Kowalski, W. (2022). Game-based sprint retrospectives: multiple action research. *Empirical Software Engineering*, 27(1):1.

[Przybyłek y Kotecka, 2017] Przybyłek, A. y Kotecka, D. (2017). Making agile retrospectives more awesome. En *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 1211–1216. IEEE.

[Robles *et al.*, 2017] Robles, G., Gonzalez-Barahona, J. M., y Fernandez, J. J. (2017). Learning outcomes and challenges in software engineering capstone projects: a systematic mapping study. En *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pp. 519–524.

[Rousseeuw, 1987] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

[Sandoval-Alfaro y Quintero-Meza, 2021] Sandoval-Alfaro, O. E. y Quintero-Meza, R. R. (2021). Application of data analytics techniques for decision making in the retrospective stage of the agile scrum methodology. En *2021 Mexican International Conference on Computer Science (ENC)*, pp. 1–8. IEEE.

[Schwaber y Sutherland, 2013] Schwaber, K. y Sutherland, J. (2013). La guía de scrum. *Scrumguides. Org*, 1:21.

[Sedelmaier y Landes, 2020] Sedelmaier, Y. y Landes, D. (2020). Analyzing challenges in software engineering capstone projects. *ICSEA 2020*, p. 145.

[Storey *et al.*, 2020] Storey, M.-A., Ernst, N. A., Williams, C., y Kalliamvakou, E. (2020). The who, what, how of software engineering research: a socio-technical framework. *Empirical Software Engineering*, 25:4097–4129.

[Törnberg, 2023] Törnberg, P. (2023). How to use large language models for text analysis. *arXiv preprint arXiv:2307.13106*.

[Vanhanen y Lehtinen, 2014] Vanhanen, J. y Lehtinen, T. O. (2014). Software engineering problems encountered by capstone project teams. *International Journal of Engineering Education*, 30(6):1461–1475.

[Vaswani *et al.*, 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., y Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Xie *et al.*, 2025] Xie, Qianqian and Chen, Qingyu and Chen, Aokun and Peng, Cheng and Hu, Yan and Lin, Fongci and Peng, Xueqing and Huang, Jimin and Zhang, Jeffrey and Keloth, Vipina and others (2025). Medical foundation large language models for comprehensive text analysis and beyond. *npj Digital Medicine*, 8(1):141.