



**Departamento de Informática**  
Universidad Técnica Federico Santa María



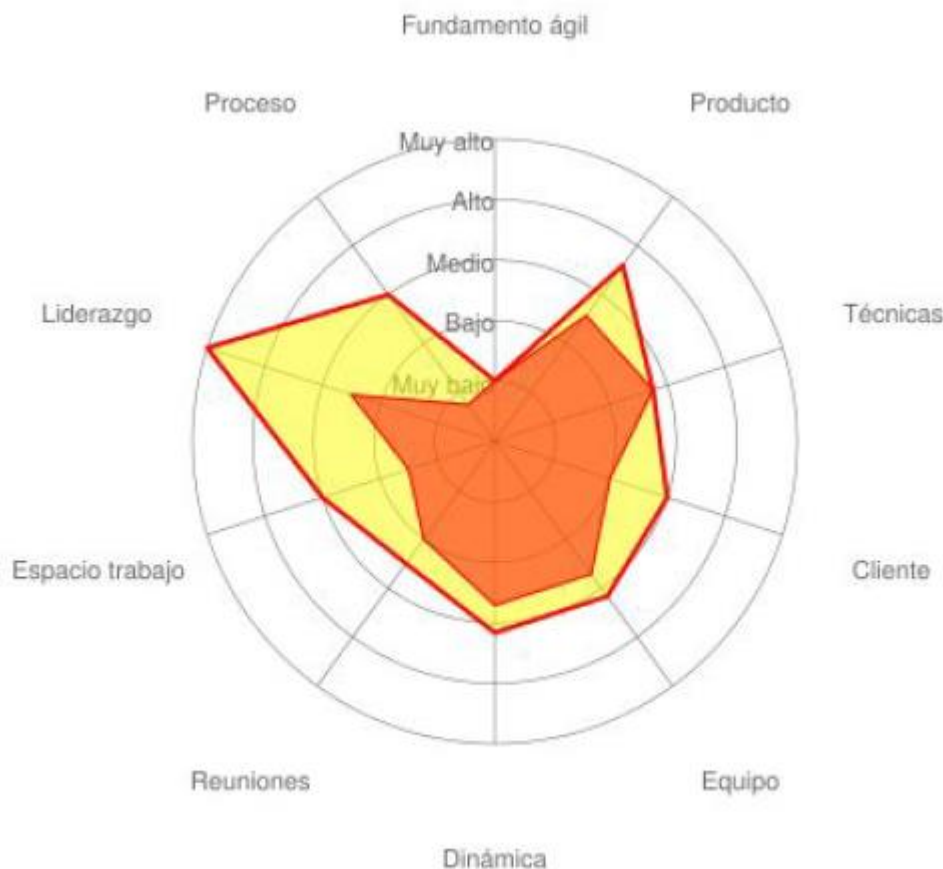
## Entregable IV

Proyecto: <KnowledgeTree>

Integrantes:

Nombres y Apellidos	Email	ROL USM
Francisco Alvial Fuentes	francisco.alvial.13@sansano.usm.cl	201373545-2
José Caimapo Sanchez	jose.caimapo.12@sansano.usm.cl	201373573-8
Luis Enrique Lemos	luis.lemos@sansano.usm.cl	201790167-5

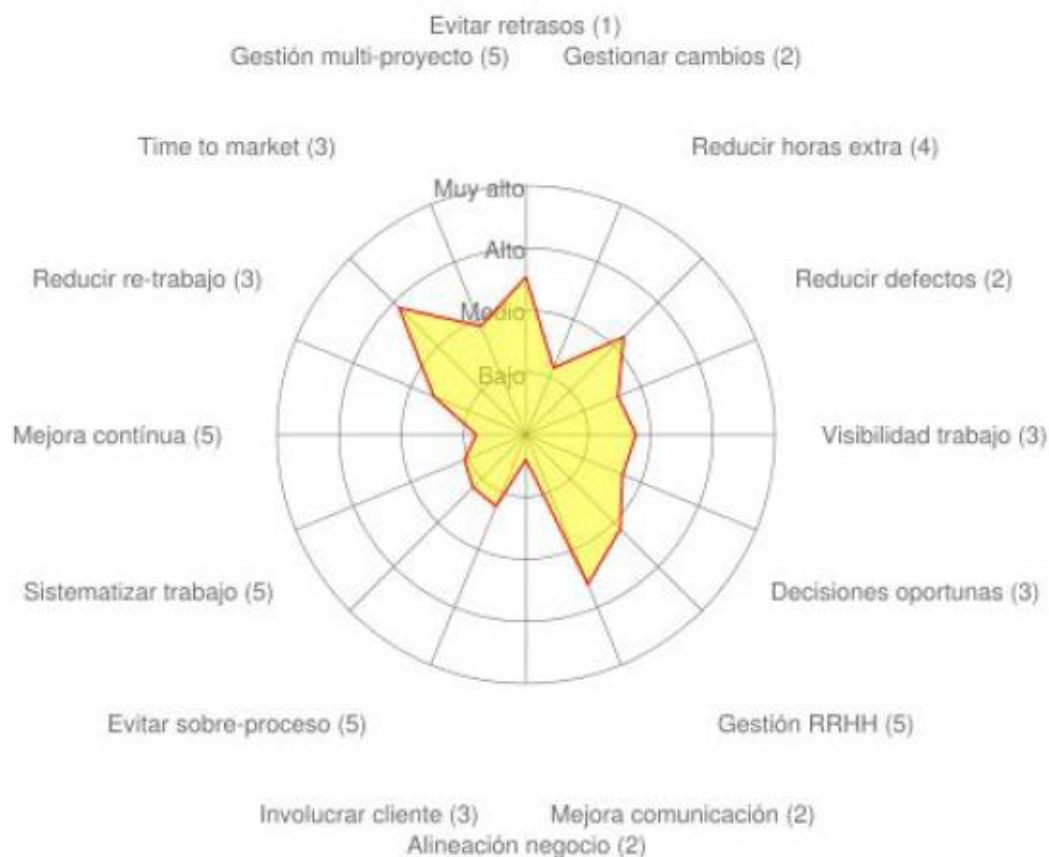
## Post-Mortem Metodológico



Del grafico de 10 dimensiones se obtuvo un bajo nivel de agilidad, debido a que no se intentó de lleno ninguna metodología ágil como tal. Se destacan las siguientes fortalezas y debilidades vistas por nosotros.

Fortalezas	Debilidades
1. Liderazgo 2. Producto 3. Técnicas	1. Reuniones 2. Espacio Trabajo

En el área de técnicas y productos esta se potenciaron dado a la toma de decisiones en la etapa de planificación que se relaciona con el liderazgo, pero flaquea en el hecho de por motivos personales que al trabajar como grupo el tiempo cara a cara entre los integrantes dado el ambiente en que se trabajó era mínimo, esto se reflejó como una debilidad, como se puede ver en el gráfico de 16 dimensiones que está seccionada por objetivos.



Como principales fortalezas del lado por el lado de objetivos se tienen:

Fortalezas	Debilidades
1. Evitar retrasos	1. Gestión de Cambios
2. Mejorar comunicación	2. Reducción de defectos
3. Decisiones oportunas	3. Visibilidad del trabajo
4. Time to market	4. Involucrar al cliente

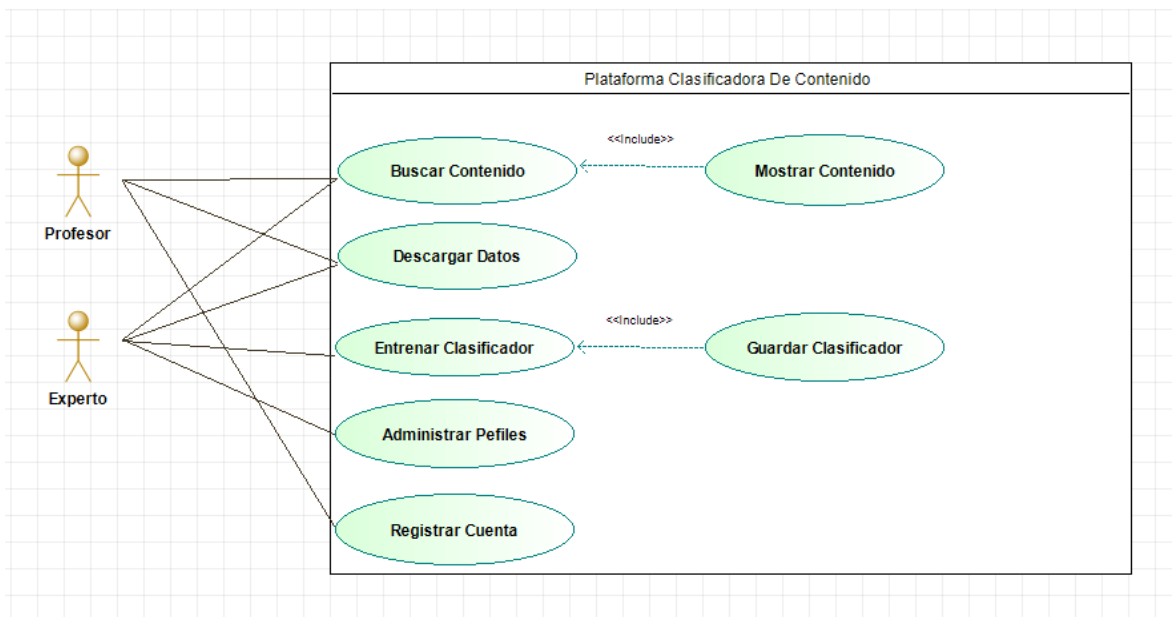
Esto se debe principalmente por la planificación previa de acotar el trabajo previsto para un período en base a la estimación y la correspondiente coherencia con la capacidad del equipo, pero se opacan con las debilidades encontradas. Las principales debilidades que se presentaron fue por la falta que espacio de trabajo en común lo que genero un pobre desempeño en la parte de objetivos de 1,2,3 y 4.

En el caso de seguir trabajando como equipo, el mayor tema es concentrarse en las presenciales. Como propuesta de mejora es la reunión de carácter obligatorio por

parte del equipo, esto compensaría grandes falencias que se detectaron por el trabajo remoto de cada uno.

## Diagrama de Casos de Uso (final)

El Diagrama de casos de uso lo entregamos en el entregable 3, así que sigue siendo el mismo que en el entregable anterior.



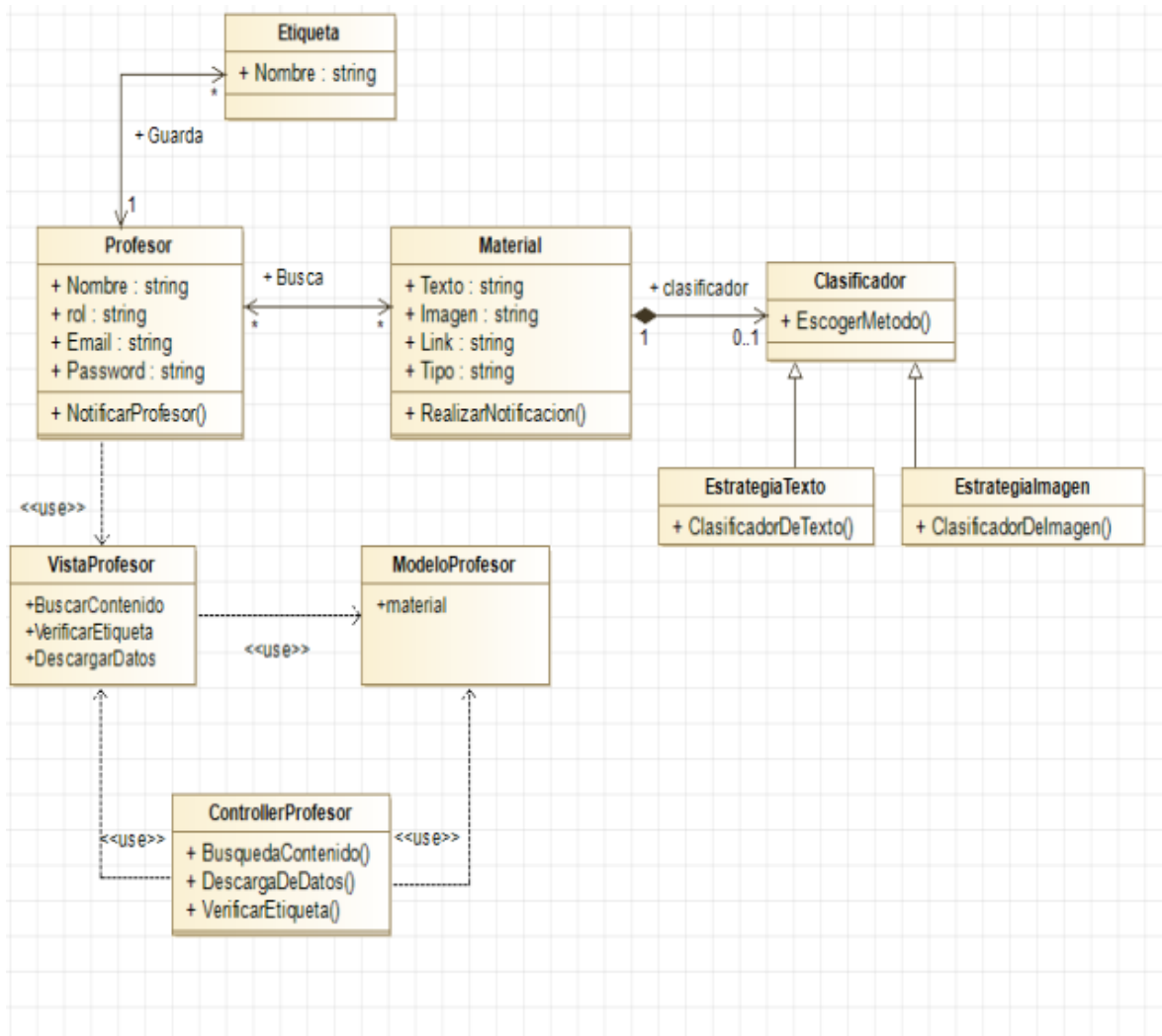
## Patrones de diseño y Frameworks (final)

Intención	Patrón de Diseño	Razonamiento
Se desea mostrar en el diagrama de clases la notificación en el sistema cuando el profesor realiza una búsqueda de algún contenido y avisar a los objetos que les interesa obtener datos sobre la búsqueda.	Observer	La clase Material debe ser capaz de avisarle al Profesor que ha recibido los textos de la búsqueda hecha, con cada link relacionado a estos. De esta forma el Profesor puede quitar los textos que no le sirvieron para que en el futuro no se usen.
Se desea mostrar en el diagrama de clases la interacción del sistema	MVC	La clase profesor debe realizar una de las interacciones (búsqueda,

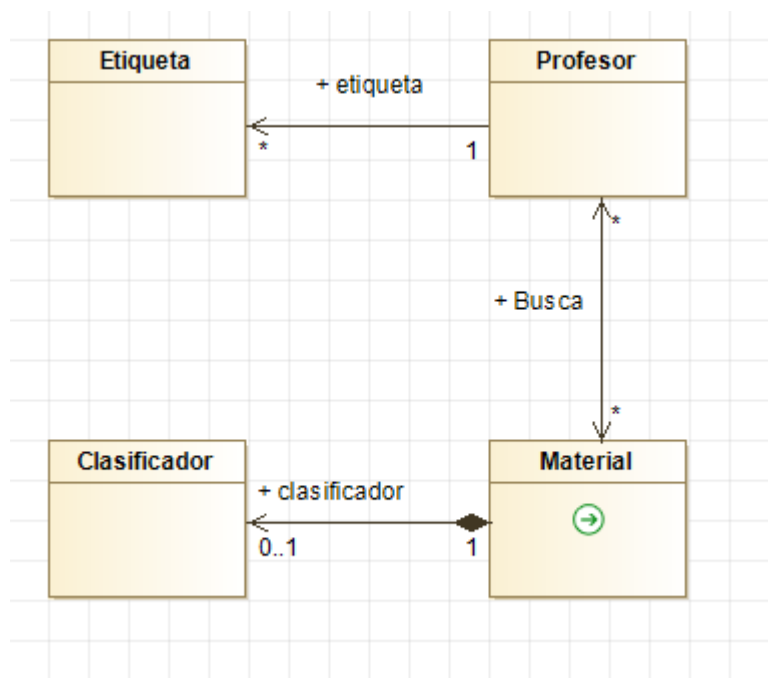
cuando el profesor realiza una búsqueda de contenido, una verificación de etiqueta y descarga de archivos.		verificación etiqueta, etc.) de la clase Vista para que ésta interactúe con sus controladores pertinentes y el modelo. Una vez terminado una interacción da inicio a una nueva.
Se propone con la intención de organizar y tener el comportamiento deseado de una parte fundamental del proyecto que es tener una función de clasificación para abordar el caso de tex	Strategy	La clase Clasificar discierne entre los inputs obtenidos para ejecutar el algoritmo apropiado de clasificación de Material en específico de texto o de imágenes, los cuales son diferentes entre sí.

## Modelo de Dominio y Diagrama de Clases (final)

### Diagrama de clases



## Modelo de Dominio



## Pruebas de Software (actualización)

Defecto encontrado	Mitigación	Resultado obtenido	Observaciones
"Buscar contenido". Se tiene que la respuesta esperada por el QA es "No se han encontrado resultados para su búsqueda, Intente nuevamente", ya que entrego una palabra sin sentido	Agregar una condición en la entrega de condición, de que si no se a encontrado el material, entregue la respuesta pedido	Al no encontrar lo buscado el programa mostrara una página donde le dice al usuario que no se encontró lo que se buscaba	ninguna
"Buscar contenido ". Se tiene que el QA espera los resultados por el texto buscado "Campo eléctrico", pero nunca la entrega, ya que el buscador no permite tilde.	No existe una manera fácil de poder realizar la mitigación, así que la solución por ahora es escribir las búsquedas sin tilde	Escribir búsquedas realizadas sin tildes.	Ninguna
Completar palabras	Esto ya está	Realizar búsqueda	Ninguno

automáticamente, ejemplo: si se escribe "Campo", que en el resultado lo complete como "Campo Eléctrico"	implementado desde antes, pero se tiene que realizar la búsqueda como mínimo 1 vez la palabra para que quede guardada	1 vez por lo menos para dejar guardado la palabra en el "auto Completar"	
--	--	--	--