

1. Filosofía Documental — Protocolo IA-Socrático USACH

Tipo: doc_inv (instrumento de investigación)

Versión: 1.0

Fecha: Mayo 2026

Propósito: Definir la filosofía, jerarquía, nomenclatura y flujo de vida de los documentos del proyecto. Este documento es la referencia operativa para cualquier persona que cree, edite o distribuya documentación del protocolo.

1.1 1. Principio rector

El .md es la fuente. Todo lo demás es derivado.

Los documentos del proyecto existen en tres niveles. El nivel 1 es lo que se edita y versiona. El nivel 2 es lo que se distribuye. El nivel 3 es lo que se archiva. Si un cambio surge en nivel 2 o 3, se sube al nivel 1 antes de propagarse.

1.2 2. Jerarquía documental

Nivel	Formato	Función	Dónde vive
1 — Fuente	.md	Documento operativo. Lo que se edita, se versiona en git, se revisa en diffs.	instrumentos/
2 — Distribución	.pdf	Documento para lectores externos. Comité de ética, colegas, revisores, alumnos. Generado desde .md vía Pandoc + XeLaTeX.	web/assets/pdf/

Nivel	Formato	Función	Dónde vive
3 — Archivo histórico	.docx	Documentos que nacieron antes de esta filosofía y no tienen fuente .md . Se conservan como trazabilidad, no se editan.	protocolo/

1.2.1 Flujo de vida de un documento

Creación: Se escribe en **.md** en instrumentos/

Edición: Se modifica el **.md** (git trackea cada cambio)

Aprobación: El IP revisa el **.md**, aprueba o pide correcciones

Distribución: Se genera **.pdf** con Pandoc + XeLaTeX → se copia a web/assets/pdf/

Archivo: El **.md** vive en instrumentos/ para siempre (fuente histórica)

1.2.2 Por qué Markdown y no Word

Criterio	.md	.docx
Versionable en git	Sí (texto plano, diff legible)	No (binario, git solo sabe que cambió)
Editable en cualquier editor	Sí	No (requiere Word o similar)
Legible sin software propietario	Sí	No
Convierte a .pdf	Sí (vía .html)	Sí (directo)
Control de cambios visible	Sí (git diff , git log)	No (requiere Word Track Changes)
Tamaño en repositorio	Mínimo	Pesado

Consecuencia: El **.docx** no aporta ventajas que el **.md** no tenga, y sí introduce problemas de versionado. Se eliminó del flujo activo.

1.2.3 Por qué el público externo no ve Markdown

El **.md** contiene: - Frontmatter técnico (**nombre:**, **tipo:**, **referencia_premisas:**) - Formato crudo sin estilos visuales - Estructura interna del repositorio (**doc_alum_**, **doc_pro_**, etc.)

Un revisor del comité de ética, un colega o un alumno no necesita ver eso. Necesita un documento limpio, con estilos, legible. Eso es el **.pdf**.

Regla: Si alguien fuera del proyecto necesita ver un documento, recibe `.pdf`. Si necesita editarlo, recibe acceso al `.md` en el repositorio.

1.3 3. Estructura de directorios

```

instrumentos/           ← NIVEL 1: Fuente (.md)
├─ doc_alum_*.md       ← Documentos que recibe el estudiante
├─ doc_pro_*.md        ← Documentos del docente/investigador en el aula
├─ doc_inv_*.md        ← Documentos del proceso investigativo
├─ clase1/ ... clase5/ ← Instrumentos por clase
│   ├─ doc_pro_GuionDocente_*.md
│   ├─ doc_pro_Protocolo_*.md
│   ├─ n1_doc_alum_*.md ← Caso técnico
│   ├─ n2_doc_alum_*.md ← Ficha pre-IA
│   ├─ n3_doc_alum_*.md ← Guía chatbot
│   └─ n4_doc_alum_*.md ← Ficha post-IA
└─ (transversales)

web/assets/pdf/        ← NIVEL 2: Distribución (.pdf)
├─ DocumentoRector_*.pdf
├─ Glosario_*.pdf
└─ (otros .pdf generados desde .md)

protocolo/             ← NIVEL 3: Archivo histórico (.docx pre-Premisas)
├─ DocumentoRector_IA_USACH_v1.1.docx
├─ Marco_Revision_Literatura_v1.2.docx
└─ Registro_Cambios_v1.0.docx
    
```

1.4 4. Nomenclatura

Todo archivo `.md` en `instrumentos/` lleva prefijo según destinatario:

Prefijo	Destinatario	Ejemplo
<code>doc_alum_</code>	Estudiante (recibe en el aula)	<code>doc_alum_Ficha1_PreAI_v1.0.md</code>
<code>doc_pro_</code>	Docente / Investigador (uso en aula)	<code>doc_pro_GuionDocente_Clase2_v1.0.md</code>
<code>doc_inv_</code>	Proceso investigativo (paper, marco)	<code>doc_inv_Premisas_Diseño_v1.1.md</code>

Regla de visibilidad: - `doc_alum_` → el estudiante lo ve - `doc_pro_` → el estudiante NUNCA lo ve - `doc_inv_` → el estudiante NUNCA lo ve

Nombre canónico de la asignatura: Laboratorio de Máquinas y Equipos Industriales (14362-0-L-1)

1.5 5. Versionado

La versión vive en **dos lugares simultáneos**:

1.5.1 5.1 Nombre del archivo

```
doc_pro_GuionDocente_Clase2_v1.0.md
      ↑           ↑
      nombre     versión
      descriptivo
```

1.5.2 5.2 Frontmatter del Markdown

```
---
nombre: Guion Docente Clase 2 v1.0
tipo: doc_pro
clase: 2
version: 1.0
fecha_emision: 2026-05-20
status: fuente_normativa
emitido_por: Ángel Royo Melgarejo (IP)
referencia_premisas: instrumentos/doc_inv_Premisas_Disenos_v1.1.md
---
```

1.5.3 5.3 Cuándo sube la versión

Cambio	Acción
Corrección menor (typo, formato)	Mismo archivo, misma versión
Cambio de contenido sin alterar método	Bump minor (v1.0 → v1.1)
Cambio metodológico o de estructura	Bump major (v1.0 → v2.0)
Nuevo documento	Empieza en v1.0

1.5.4 5.4 Pareo obligatorio

Todo cambio metodológico en `Protocolo_ClaseN` debe revisarse contra `GuionDocente_Clase N` y viceversa. Si uno cambia y el otro no, hay inconsistencia.

1.6 6. Estados de un documento

Estado	Significado
fuelle_normativa	Documento vigente, aprobado por el IP
borrador	En desarrollo, no aprobado
superseded	Reemplazado por versión mayor. Se conserva como trazabilidad
deprecated	Ya no aplica. Se conserva por historia
histórico	Solo existe como <code>.docx</code> pre-Premisas. No se edita

1.7 7. Roles y responsabilidades

Rol	Puede editar	Puede aprobar	Puede distribuir
Investigador Principal (IP)	Todos	Todos	Todos
Agente desarrollador	Todos (bajo instrucción del IP)	Ninguno	Genera <code>.pdf</code> , actualiza web
Docente ejecutor	Ninguno	Ninguno	Usa los instrumentos en el aula
Estudiante	Ninguno	Ninguno	Recibe <code>doc_alum_</code>

Regla: Nadie edita un documento sin que el IP lo autorice. El agente desarrollador ejecuta instrucciones, no decide contenido.

1.8 7b. Principios de puesta en escena pedagógica

***Principio rector:** La IA no enseña máquinas por el estudiante; la IA obliga al estudiante a pensar como técnico frente a una máquina.*

1. La clase se presenta como laboratorio de diagnóstico industrial, no como demostración tecnológica. El estudiante no debe sentir que participa en una prueba de software: debe sentir que enfrenta una instalación realista con síntomas observables, sensores imperfectos, operadores humanos, datos incompletos, restricciones de tiempo y una decisión que tendrá consecuencias.
2. El estudiante entra primero al problema técnico: una máquina, instalación, sensor, falla, operador, restricción o decisión realista. Solo después aparece la IA como herramienta subordinada para presionar el razonamiento.
3. El protagonista de la experiencia no es el chat, el admin, el modelo, el webhook ni la base de datos. El protagonista es el sistema industrial bajo incertidumbre. La infraestructura digital existe para capturar, presionar y analizar razonamiento, pero debe permanecer invisible para el estudiante salvo cuando sea estrictamente necesario operar la actividad.
4. El chat no entrega respuestas ni decora productos. Su función pedagógica es detectar razonamiento incompleto: hipótesis vagas, variables no consideradas, datos faltantes, confusión entre síntoma y causa, uso acrítico de sensores o decisiones sin riesgo explícito.
5. La evidencia relevante del piloto no es que el estudiante use IA, sino que cambie la calidad de su razonamiento técnico: que formule mejores hipótesis, identifique variables críticas, reconozca incertidumbre, justifique decisiones y pueda defenderlas frente a operación, mantenimiento y seguridad.
6. Toda clase debe partir desde una máquina, instalación, sensor, falla, operador, restricción o decisión realista. La IA debe ayudar a descubrir razonamiento incompleto, no a reemplazarlo.
7. El éxito del piloto no es que el estudiante diga “estuvo bueno el chatbot”, sino “aprendí a mirar una máquina distinto”.

Regla documental: En materiales visibles para estudiantes se evita presentar la arquitectura tecnológica como contenido de aula. Términos como admin, dashboard, n8n, webhook, base de datos, LLM, MVP, system prompt, modo PLAN y modo BUILD pertenecen a documentos docentes, técnicos o metodológicos. En la escena de aprendizaje se traducen como contraste de hipótesis, revisión profesional, comité de crisis, decisión defendible o transferencia autónoma.

1.9 8. Distribución

1.9.1 8.1 Para el público externo

- **Formato:** .pdf
- **Ubicación:** web/assets/pdf/
- **Acceso:** página de descargas en la web ([web/descargas.html](#))
- **Regla:** nunca se distribuye .md ni .docx a externos.

1.9.2 8.2 Para el equipo interno

- **Formato:** `.md` (fuente directa)
- **Ubicación:** `instrumentos/`
- **Acceso:** repositorio git
- **Regla:** siempre trabajar sobre el `.md`, nunca sobre el `.pdf`, HTML o assets derivados.

1.9.3 8.3 Flujo de generación de PDF

Flujo vigente para documentos activos:

```
pandoc instrumentos/<fuente>.md \
-o pdf/output/<destino>.pdf \
--template=pdf/templates/academico.latex \
--pdf-engine=xelatex
```

Luego el PDF verificado se copia a `web/assets/pdf/` en la carpeta correspondiente. El PDF es derivado; si hay una corrección, se modifica primero el `.md` fuente y se regenera.

No usar como flujo oficial: PDF24, impresión del navegador, Playwright/browser print ni `.md` → `.html` → `.pdf` para documentos activos.

1.9.4 8.4 Especificación LaTeX para PDFs

Los PDFs derivados del proyecto deben verse como documentos académicos LaTeX de alta calidad, no como HTML impreso. Usar tipografía clásica LaTeX basada en Latin Modern / Computer Modern, tablas con `booktabs`, márgenes amplios, jerarquía sobria, citas limpias y portada institucional austera.

***Skill de implementación:** `.agents/skills/pdf-latex-lujo/SKILL.md` — contiene el flujo Pandoc/XeLaTeX, criterio tipográfico, template y verificación.*

Reglas obligatorias:

Elemento	Regla
Fuente primaria	<code>.md</code> en <code>instrumentos/</code>
Motor PDF	Pandoc + XeLaTeX
Template	<code>pdf/templates/academico.latex</code>
Tablas	<code>booktabs</code> , sin estilo HTML impreso
Identidad institucional	Universidad de Santiago de Chile — Facultad de Ingeniería — Departamento de Ingeniería Industrial

Elemento	Regla
Pie de página	Facultad de Ingeniería — Departamento de Ingeniería Industrial / Profesor Ángel Royo - www.angelroyo.com
Numeración	N de M
Verificación	compilar sin errores, abrir PDF y revisar portada, tildes/ñ, tablas, saltos, citas, encabezado/pie

Regla: si el PDF no cumple el estándar LaTeX académico, no se publica.

1.10 9. Trazabilidad

1.10.1 9.1 Registro de Cambios

protocolo/Registro_Cambios_v1.0.docx documenta: - Qué documento cambió - De qué versión a qué versión - Qué se modificó - Quién lo autorizó - Fecha

1.10.2 9.2 Git como trazabilidad operativa

Cada commit en git es un punto de trazabilidad. El mensaje del commit describe qué cambió. `git log` permite reconstruir la historia completa de cualquier archivo.

Regla: Un commit por fase lógica. No mezclar reescrituras completas con ajustes menores.

1.11 10. Jerarquía de autoridad

Nivel	Documento	Función	Cuándo cambia
1 — Por qué	protocolo/Documento Rector_IA_USACH_v1.1.docx	Problema, hipótesis, marco general	Solo si cambia la investigación fundamentalmente
2 — Qué y cómo	instrumentos/doc_in v_Premisas_Disenio_v 1.1.md	Fuente de verdad del diseño: P1-P9, DD_1-DD_39	Solo tras debate explícito con el IP

Nivel	Documento	Función	Cuándo cambia
3 — Implementación	Todos los demás <code>.md</code>	Ejecución concreta del nivel 2	Se ajusta libremente mientras sea consistente con nivel 2

Regla fundamental: Si un instrumento de nivel 3 contradice una decisión DD_1-DD_39, se corrige el instrumento. Si la decisión resulta inviable, se abre debate antes de modificar nivel 2.

1.12 11. Preguntas frecuentes

¿Puedo editar un `.pdf` directamente?

No. El `.pdf` es derivado. Si necesita cambios, se edita el `.md` fuente y se regenera el `.pdf`.

¿Qué hago si encuentro un error en un `.docx` de protocolo/?

Ese archivo es histórico pre-Premisas. No se edita. Si el contenido es relevante, se crea un `.md` nuevo en `instrumentos/` con la versión corregida.

¿Cómo sé cuál es la versión vigente de un documento?

Consultar la tabla de versiones en `AGENTS.md`. Siempre es la versión más alta del mismo nombre base.

¿Puedo crear un documento sin prefijo `doc_alum_ / doc_pro_ / doc_inv_`?

No. Todo `.md` en `instrumentos/` debe llevar prefijo. Es la convención que permite identificar el destinatario sin abrir el archivo.

Filosofía Documental v1.0 · Protocolo IA-Socrático · USACH · Mayo 2026