ANEXO · TAREAS DEL ALUMNADO

ANTES DE LA PRIMERA PARTIDA: MANUALES DE ADMINISTRACIÓN, DATOS Y TUTORIAL

Nombre: _____ Fecha: _____

ANTES DE EMPEZAR · TRES MANUALES PARA QUIEN LA PONE EN MARCHA

Instalación (qué necesito, qué escribo, cómo sé que ha funcionado) · **Configuración** (cada opción: nombre, dónde está, valores, valor por defecto, efecto) · **Administración** (copia de seguridad y restauración, actualizar, reiniciar, qué hacer si se rompe). Reglas: versiones siempre, cada comando solo en su línea dentro de un bloque de código, y la instalación acaba con una comprobación.

1 El README de tu UT3 dice «npm install y funciona». ¿Cuál de los tres manuales es y qué le falta?

TAREA 1 · TU PÁGINA DE DATOS (12 MIN)

Calentamiento en Figma (4 min). Duplica el archivo 09022027 · Data structure diagram. Marco **2**: sustituye cada «?» rojo por un tipo y escribe la cardinalidad en la línea. Marco **3**: completa dos filas de la tabla (`tags` y `updatedAt`). Las reglas están en el marco 1 y en la app: intenta romperlas. ¿Qué pasa con seis etiquetas? ¿Y con un título vacío?

Tu app (8 min). Crea `docs/data.md` en tu proyecto:

OK	HAZ	ESTÁ HECHO CUANDO…
☐	Dónde y cómo: el almacenamiento, las claves o archivos, el formato y la versión de la estructura.	Alguien sabe dónde mirar sin leer tu código.
☐	El diagrama con un `erDiagram` de Mermaid (ejemplo abajo).	El diagrama se ve en el sitio.
☐	La tabla de tu entidad principal con cinco columnas: **Field · Type · Required · Rules · Example**.	Ninguna celda de Rules vacía sin pensarla.
☐	Un ejemplo real, copiado de la app, en un bloque de código.	El ejemplo sale de DevTools o del archivo, no inventado.
☐	La página en `nav`.	`zensical build` sin errores.

````
```mermaid
erDiagram
    NOTE {
        string id PK
        string title
        boolean pinned
    }
    TAG {
        string name
    }
    NOTE ||--o{ TAG : "tags (0 to 5)"
```
````

`||--o{` se lee: **una** nota tiene **cero o más** etiquetas.

Si algo falla:

- **Tipos del lenguaje, no de los datos** («Date»). En disco una fecha es una cadena con un formato: di cuál (ISO 8601, `2027-02-09T08:10:42.120Z`).
- **La columna Rules está vacía.** ¿Puede estar vacío? ¿Se puede repetir? ¿Cuál es el máximo? Si la app lo comprueba, es una regla.
- **El diagrama no se dibuja.** Mermaid es estricto: nombres de entidad sin espacios, un campo por línea, primero el tipo.

Si acabas pronto: el marco 4 de Figma para tu app y la sección «límites y copia de seguridad».

TAREA 2 · TU TUTORIAL (10 MIN)

Las cinco reglas de un tutorial:

1. **Di qué tendrán al final**, en la primera frase, y cuánto tarda.
2. **Un ejemplo concreto**: di exactamente qué escribir («escribe `Weekend shopping`»), no «escribe un título».
3. **Cada paso acaba con «You should see…»**
4. **Sin opciones y sin explicaciones.** Nada de «también puedes…» ni «esto funciona porque…». Enlaza al manual.
5. **Funciona siempre**, desde la app vacía. Pruébalo desde cero antes de publicarlo.

OK	HAZ	ESTÁ HECHO CUANDO…
☐	Crea `docs/tutorial.md` y añádelo al `nav`.	Sale en el menú del sitio.
☐	Primera frase con el resultado y el tiempo.	El lector sabe qué va a conseguir.
☐	**Cinco pasos**, cada uno con una instrucción exacta y un «You should see».	Cada paso dice qué aparece y dónde.
☐	Última línea: qué ha aprendido y adónde ir después.	Enlaza a la siguiente página.

Si algo falla:

- **Es el manual otra vez** («To add a note, press Add»). ¿Qué escribo **exactamente**?
- **El paso 3 depende de datos que tú tienes y el lector no.** Un tutorial empieza desde la app vacía.
- **Diez pasos.** Cinco. Un tutorial terminado vale más que uno completo.

Si acabas pronto: borra los datos de tu app y sigue tu propio tutorial al pie de la letra, desde cero.

ENTREGA

Commit y push de `docs/data.md` y `docs/tutorial.md` (en `nav`, con el sitio compilando). En Classroom: el enlace al repositorio y el enlace de Figma.

PARA CASA

- Escribe `docs/admin.md` para tu app con estas secciones: requisitos con versiones, pasos de instalación, la comprobación, la tabla de opciones (Option · Where · Values · Default · Effect), copia de seguridad y restauración. Añádelo al `nav`.
- Reconstruye todo desde una copia limpia: `npm run docs:shots`, `zensical build --clean`, `npm run docs:pdf`. Si solo funciona en tu PC, tu manual de instalación está mal.
- Antes del viernes: comprueba que `zensical`, `pandoc`, `typst` y el navegador de Playwright funcionan en el PC que traerás al examen.

Examen, viernes 12/02, 2 h: una parte escrita corta y práctica sobre un proyecto NoteBox que recibirás (página de manual con captura, sitio y PDF; una ayuda contextual diseñada en Figma e implementada; la página de datos; un tutorial corto). Tu propio PC y la documentación oficial de las herramientas; **sin IA ni mensajería**.