

---

# **TeachBook (Plantilla Ciencias Químicas)**

**Álvaro Lozano Murciego y André Sales Mendes (USAL)**

**Feb 10, 2026**



# CONTENTS

<b>I Tutorial</b>	<b>3</b>
<b>1 1. ¿Qué es un TeachBook?</b>	<b>5</b>
1.1 Ventajas para la docencia . . . . .	5
1.2 Tecnología subyacente . . . . .	5
<b>2 2. Flujo de Trabajo</b>	<b>7</b>
2.1 Estructura de archivos . . . . .	7
<b>3 3. Edición con IA</b>	<b>9</b>
3.1 Cómo pedir ayuda a la IA . . . . .	9
3.2 Ejemplo de Prompt . . . . .	9
<b>4 4. Publicación Web</b>	<b>11</b>
4.1 Pasos para publicar . . . . .	11
<b>II Ejemplos por Grado</b>	<b>13</b>
<b>5 Grado en Física</b>	<b>15</b>
5.1 Ejemplo de Física: Oscilador Armónico . . . . .	15
<b>6 Grado en Matemáticas</b>	<b>17</b>
6.1 Ejemplo de Matemáticas: Cálculo Simbólico . . . . .	17
<b>7 Grado en Estadística</b>	<b>19</b>
7.1 Ejemplo de Estadística: Generación de Datos . . . . .	19
<b>III Información</b>	<b>21</b>
<b>8 Acerca de</b>	<b>23</b>
8.1 Autores . . . . .	23
8.2 Contexto . . . . .	23
8.3 Año . . . . .	23
<b>9 Licencias</b>	<b>25</b>
9.1 Contenidos . . . . .	25
9.2 Código . . . . .	25
9.3 Atribuciones . . . . .	25
<b>10 Cómo citar</b>	<b>27</b>

10.1	Cita textual . . . . .	27
10.2	BibTeX . . . . .	27
10.3	DOI . . . . .	27

Bienvenido a la plantilla **TeachBook Ciencias Template**.

## ¿Qué es esto?

Es una plantilla diseñada para que el profesorado de la **Facultad de Ciencias de la USAL** pueda crear libros docentes interactivos de forma sencilla.

## Contenido

En este libro encontrarás:

- *Tutoriales* para aprender a usar la plantilla
- *Ejemplos por Grado* para ver casos reales
- Información sobre *cómo citar* y *licencias*

---

**Note:** Este proyecto está diseñado para ser usado con **VS Code** y asistentes de IA.

---



# **Part I**

# **Tutorial**



## 1. ¿QUÉ ES UN TEACHBOOK?

Un **TeachBook** es un libro digital interactivo que combina:

- Texto explicativo
- Código ejecutable (Python, R, Julia...)
- Gráficos interactivos
- Ecuaciones matemáticas

### 1.1 Ventajas para la docencia

1. **Accesibilidad:** Disponible en web desde cualquier dispositivo.
2. **Interactividad:** Los estudiantes pueden modificar el código y ver resultados.
3. **Mantenibilidad:** Fácil de actualizar y corregir.
4. **Reproducibilidad:** Todo el contenido se genera desde el código fuente.

### 1.2 Tecnología subyacente

Este proyecto utiliza [Jupyter Book](#), una herramienta estándar en la comunidad científica.



## 2. FLUJO DE TRABAJO

Este es el ciclo de vida recomendado para tu TeachBook:

1. **Escribir contenido** en archivos Markdown (.md) o Notebooks (.ipynb).
2. **Previsualizar** los cambios localmente (opcional, pero recomendado).
3. **Guardar versiones** (commit) con Git.
4. **Publicar** (push) a GitHub para que la web se actualice automáticamente.

### 2.1 Estructura de archivos

El contenido se organiza en la carpeta book/es/.

- intro.md: Página principal.
- \_toc.yml: Índice del libro.
- \_config.yml: Configuración general.



## 3. EDICIÓN CON IA

Este template está diseñado para usarse con asistentes de IA como **GitHub Copilot** o **Antigravity**.

### 3.1 Cómo pedir ayuda a la IA

Puedes pedirle cosas como:

- “Explícame este código de Python”
- “Genera un gráfico de una función seno”
- “Revisa la ortografía de este párrafo”
- “Añade una ecuación matemática en formato LaTeX”

### 3.2 Ejemplo de Prompt

“Crea una celda de código que grafique la distribución normal estándar usando matplotlib.”

La IA te dará el código y tú solo tienes que ejecutarlo o incluirlo en tu libro.



## 4. PUBLICACIÓN WEB

La publicación se realiza de forma automática a través de **GitHub Pages**.

### 4.1 Pasos para publicar

1. Asegúrate de que tu repositorio es público (o tienes acceso a GitHub Pages en privado).
2. Haz cambios en tu contenido.
3. Guarda los cambios un **Commit** y haz **Push** a la rama principal (`main`).
4. Una “Acción de GitHub” se ejecutará automáticamente y construirá tu libro.
5. En unos minutos, verás tu libro actualizado en la URL de tu repositorio (configurada en *Settings > Pages*).



## **Part II**

# **Ejemplos por Grado**



## GRADO EN FÍSICA

En esta sección encontrarás ejemplos adaptados a asignaturas del Grado en Física. El objetivo es mostrar cómo se pueden integrar:

- Fórmulas matemáticas complejas
- Gráficos de simulaciones físicas
- Código de análisis de datos experimentales

### 5.1 Ejemplo de Física: Oscilador Armónico

Este notebook simula un oscilador armónico simple.

```
import numpy as np
import matplotlib.pyplot as plt

# Parámetros
k = 1.0 # Constante del resorte
m = 1.0 # Masa
omega = np.sqrt(k / m)
t = np.linspace(0, 20, 100)

# Posición
x = np.cos(omega * t)

# Gráfico
plt.figure(figsize=(8, 4))
plt.plot(t, x)
plt.title('Oscilador Armónico Simple')
plt.xlabel('Tiempo (s)')
plt.ylabel('Posición (m)')
plt.grid(True)
plt.show()
```



## GRADO EN MATEMÁTICAS

Ejemplos para asignaturas de matemáticas. Aquí se prioriza:

- Rigor en la notación matemática (LaTeX)
- Demostraciones visuales
- Algoritmos simbólicos (SymPy)

### 6.1 Ejemplo de Matemáticas: Cálculo Simbólico

Vamos a usar SymPy para derivar e integrar funciones.

```
from sympy import symbols, diff, integrate, sin, exp

x = symbols('x')
f = exp(-x) * sin(x)

# Derivada
df = diff(f, x)
print(f"Derivada: {df}")

# Integral indefinida
int_f = integrate(f, x)
print(f"Integral: {int_f}")
```



## GRADO EN ESTADÍSTICA

Ejemplos orientados al análisis de datos y probabilidad. Destacamos:

- Manejo de DataFrames (Pandas)
- Visualización estadística (Seaborn)
- Modelos probabilísticos

### 7.1 Ejemplo de Estadística: Generación de Datos

Generamos una distribución normal y visualizamos su histograma.

```
import numpy as np
import matplotlib.pyplot as plt

data = np.random.normal(0, 1, 1000)

plt.hist(data, bins=30, alpha=0.7, color='green', density=True)
plt.title('Histograma de Distribución Normal')
plt.show()
```



# **Part III**

# **Información**



## ACERCA DE

### 8.1 Autores

**Facultad de Ciencias** Universidad de Salamanca (USAL)

### 8.2 Contexto

Este libro forma parte del proyecto de innovación docente para la integración de herramientas digitales en la enseñanza de las ciencias.

### 8.3 Año

2025



## LICENCIAS

### 9.1 Contenidos

Todo el contenido de texto e imágenes (salvo que se indique lo contrario) se distribuye bajo licencia **Creative Commons Atribución 4.0 Internacional (CC BY 4.0)**.

### 9.2 Código

Los ejemplos de código fuente se distribuyen bajo licencia **MIT**.

### 9.3 Atribuciones

Este proyecto utiliza:

- Jupyter Book
- The Turing Way (como inspiración)



## CÓMO CITAR

Si utilizas este material, por favor cítalo de la siguiente manera:

### 10.1 Cita textual

Facultad de Ciencias (2025). *TeachBook Sciences Template*. Universidad de Salamanca. Disponible en:  
[URL del repositorio]

### 10.2 BibTeX

```
@book{teachbook_sciences_2025,
  author = {Facultad de Ciencias},
  title = {TeachBook Sciences Template},
  year = {2025},
  publisher = {Universidad de Salamanca},
  url = {https://github.com/usuario/repo}
}
```

### 10.3 DOI

(Pendiente de asignación en Zenodo)