

QUARTO_2025

Norah Jones

Invalid Date

Table of contents

| | |
|--|-----------|
| Prefacio | 3 |
| 1 Introducción a Quarto | 4 |
| 2 Usos Principales de Quarto | 5 |
| 3 Creación de un Proyecto de Tipo Libro con Quarto | 6 |
| 3.1 Pasos para crear un nuevo proyecto libro en Visual Studio Code | 6 |
| 3.2 Estructura básica del archivo <code>_quarto.yml</code> del libro | 7 |
| 4 Agregar el Repositorio Remoto en Git | 8 |
| 4.1 Comando para agregar el repositorio remoto | 8 |
| 4.2 Verificar el remoto agregado | 8 |
| 5 Publicación Automática en GitHub Pages con Quarto | 9 |
| 5.1 Configuración para publicar en GitHub Pages desde <code>_quarto.yml</code> | 9 |
| 5.2 Pasos para publicar el libro en GitHub Pages | 9 |
| 6 Sincronización del Proyecto Local con el Repositorio en GitHub | 11 |
| 6.1 Pasos para sincronizar la copia local con GitHub usando Git | 11 |
| 7 Buenas Prácticas y Recomendaciones | 13 |
| 8 Referencias | 14 |
| 9 Summary | 15 |
| References | 16 |

Prefacio

Este manual ha sido creado con el propósito de proporcionar una guía integral, técnica y detallada sobre el uso de Quarto, una tecnología emergente y poderosa para la creación de documentos científicos, técnicos y educativos de alta calidad. Está dirigido a desarrolladores, investigadores, educadores y profesionales del área de la documentación técnica que busquen optimizar la generación, gestión y publicación de sus contenidos.

En las páginas siguientes se exploran los fundamentos de Quarto: qué es, para qué se usa y cuáles son sus principales ventajas en la producción de textos reproducibles e interactivos. Se ofrece una explicación técnica formal orientada a la creación de proyectos de tipo libro, con un enfoque metodológico que abarca desde la estructura inicial en Visual Studio Code hasta la publicación automática en GitHub Pages, facilitando así la diseminación eficiente y profesional de la información.

Este manual también aborda la integración esencial con sistemas de control de versiones mediante Git y GitHub, enfatizando el procedimiento para sincronizar el repositorio local con el repositorio remoto, un paso clave para el mantenimiento y colaboración en proyectos modernos.

La intención es que esta obra sirva tanto como material de referencia para principiantes que se inician en Quarto, como para usuarios avanzados que buscan automatizar y perfeccionar sus flujos de trabajo de documentación técnica.

Finalmente, se invita al lector a explorar y aplicar los contenidos aquí expuestos, con la confianza de que facilitarán un enfoque profesional y actualizado en la elaboración y publicación de materiales técnicos en el ámbito académico, empresarial o científico.

Profe Chris Jara
Cuenca, Ecuador
Docente de Informática

1 Introducción a Quarto

Quarto es un sistema moderno y versátil para la creación de documentos técnicos y científicos, que permite integrar texto, código y resultados de análisis en un solo flujo reproducible. Está diseñado para soportar múltiples lenguajes de programación como R, Python, Julia y Observable JavaScript, y se utiliza ampliamente para elaborar reportes, libros, presentaciones, blogs y sitios web académicos o profesionales.

El sistema aprovecha la sintaxis Markdown enriquecida y extiende sus capacidades para incluir gráficos, ecuaciones en LaTeX, diagramas, tablas, y fragmentos de código ejecutables, brindando un entorno ideal para documentar proyectos científicos, educativos y de software con alta calidad.

2 Usos Principales de Quarto

- **Documentación reproducible:** Integración directa de código y resultados con la narrativa texto.
 - **Publicaciones académicas:** Creación de artículos científicos o tesis con soporte para referencias y formatos estándar.
 - **Libros técnicos:** Unificación de múltiples capítulos, manejo de bibliografía, tablas de contenido y referencias cruzadas.
 - **Sitios web y blogs:** Publicación multimedia con facilidades para estilos, temas y desplegado en servidores o GitHub Pages.
 - **Presentaciones interactivas:** Creación de diapositivas con código incrustado y visualizaciones.
-

3 Creación de un Proyecto de Tipo Libro con Quarto

Un proyecto libro en Quarto es una colección organizada de documentos (*.qmd*), cada uno representando capítulos o secciones, que se compila en un único manuscrito con múltiples formatos de salida (HTML, PDF, EPUB, entre otros).

3.1 Pasos para crear un nuevo proyecto libro en Visual Studio Code

1. Abrir Visual Studio Code (VS Code).

2. Abrir la terminal de comandos

- Presionar Ctrl + P (o Cmd + P Mac) y ejecutar el comando:

```
Quarto: Create Project
```

3. Seleccionar el tipo de proyecto

- Elegir la opción:

```
Book Project
```

4. Seleccionar la carpeta donde se creará el proyecto

- Navegar a la ruta deseada en tu sistema de archivos.

5. Nombrar la carpeta del proyecto

- Ejemplo: `SistemasOperativos_Tercero`.

6. Quarto generará automáticamente la estructura base del libro, que incluye:

- Archivos esenciales:
 - `_quarto.yml` (configuración general del libro).
 - `index.qmd` (página principal o portada).

- Otros capítulos predefinidos (`intro.qmd`, `summary.qmd`, etc.)
- Directorios para recursos auxiliares (imágenes, referencias bibliográficas).

3.2 Estructura básica del archivo `_quarto.yml` del libro

`project:` `type:` `book`

`book:` `title:` “Sistemas Operativos - Segundo Nivel” `author:` “Profe Chris Jara” `date:` “2025-09-22” `chapters:` - `index.qmd` - `intro.qmd` - `chapter1.qmd` - `summary.qmd` `bibliography:` `references.bib`

`format:` `html:` `theme:` `cosmo` `toc:` `true` `pdf:` `documentclass:` `scrreprt`

Este archivo controla desde el título y autor hasta el orden y contenido del libro, así como los formatos de salida y su apariencia.

4 Agregar el Repositorio Remoto en Git

Antes de sincronizar tu proyecto local con GitHub, es necesario vincular el repositorio remoto con el proyecto local usando Git.

4.1 Comando para agregar el repositorio remoto

Desde la terminal integrada en VS Code, estando en la carpeta raíz del proyecto, ejecuta:

```
git remote add origin https://github.com/profechrisjara/SistemasOperativos_Tercero.git
```

- `origin` es el nombre convencional para el repositorio remoto principal.
- La URL debe corresponder al repositorio creado en GitHub donde deseas subir tu proyecto.

4.2 Verificar el remoto agregado

Para confirmar que el control remoto fue agregado correctamente, usa:

```
git remote -v
```

Debería mostrar algo similar a:

```
origin https://github.com/profechrisjara/SistemasOperativos_Tercero.git (fetch)  
origin https://github.com/profechrisjara/SistemasOperativos_Tercero.git (push)
```

5 Publicación Automática en GitHub Pages con Quarto

Quarto permite automatizar la publicación del libro directamente en GitHub Pages, facilitando que el contenido generado en HTML esté disponible en la web con solo un comando.

5.1 Configuración para publicar en GitHub Pages desde `_quarto.yml`

```
publish:
  type: github
  method: branch
  repository: https://github.com/profechrisjara/SistemasOperativos_Tercero.git
  branch: gh-pages
  pre-render: true
```

- `type: github` indica que se publicará en GitHub.
- `method: branch` usa un branch dedicado (`gh-pages`) para la publicación web.
- `repository` es el repositorio remoto en GitHub donde se publicará el contenido.
- `branch` especifica la rama donde se despliega la página.
- `pre-render: true` indica que Quarto debe renderizar el contenido automáticamente antes de publicar.

5.2 Pasos para publicar el libro en GitHub Pages

1. Asegúrate de estar en la carpeta raíz del proyecto:

```
cd /ruta/a/SistemasOperativos_Tercero
```

2. Ejecuta el comando de publicación de Quarto:

`quarto publish`

3. Quarto renderiza el libro y sincroniza la carpeta generada (`_book/`) con la rama `gh-pages` del repositorio remoto.
4. Configura GitHub Pages en el repositorio:
 - En GitHub, accede a **Settings > Pages**.
 - Selecciona como fuente la rama `gh-pages`.
 - Guarda la configuración para que el sitio quede publicado.
5. La URL pública suele ser:

`https://profechrisjara.github.io/SistemasOperativos_Tercero/`

6 Sincronización del Proyecto Local con el Repositorio en GitHub

Mantener sincronizado un proyecto local de Quarto con su repositorio remoto en GitHub es fundamental para asegurar que los cambios realizados localmente se reflejen en el repositorio en línea y que los cambios remotos, si los hubiera, se integren correctamente en el entorno de trabajo local.

6.1 Pasos para sincronizar la copia local con GitHub usando Git

1. Abrir la terminal integrada de VS Code y navegar a la carpeta raíz de tu proyecto:

```
cd /ruta/a/SistemasOperativos_Tercero
```

2. Agregar el repositorio remoto (si no está agregado):

```
git remote add origin https://github.com/profechrisjara/SistemasOperativos_Tercero.git
```

Nota: Si aparece el mensaje de error fatal: remote origin already exists., usa

```
git remote set-url origin https://github.com/profechrisjara/SistemasOperativos_Tercero.git
```

para actualizar la URL del remoto.

3. Agregar cambios al área de preparación (stage):

```
git add .
```

Esto recoge todos los archivos modificados y nuevos para incluirlos en el próximo commit.

4. Crear un commit con un mensaje descriptivo:

```
git commit -m "Descripción breve de los cambios"
```

Esto guarda los cambios localmente en el historial de Git.

5. Enviar los cambios al repositorio remoto (GitHub):

```
git push -u origin main
```

- Si la rama principal local se llama **master**, cambiar **main** por **master**.

6. Actualizar los cambios realizados en el repositorio remoto al entorno local:

Para traer cambios realizados por otros colaboradores o desde GitHub:

```
git pull origin main
```

7. Verificar el repositorio remoto configurado:

Para asegurarte que tu proyecto local está apuntando al repositorio correcto:

```
git remote -v
```

7 Buenas Prácticas y Recomendaciones

- Realizar commits con mensajes claros y frecuentes para facilitar el seguimiento de cambios.
 - Sincronizar periódicamente con el repositorio remoto para evitar conflictos.
 - Configurar autenticación con token personal (PAT) o clave SSH para operaciones push/pull seguras.
 - Aprovechar la previsualización y el render automático de Quarto para validación antes de publicar o subir.
-

8 Referencias

- Documentación oficial de Quarto: <https://quarto.org/docs/books/>
- Publicación en GitHub Pages con Quarto: <https://quarto.org/docs/publishing/github-pages/>
- Git - Comandos para gestión de remotos: <https://docs.github.com/en/get-started/git-basics/managing-remote-repositories>
- GitHub Docs - Sincronización de ramas: <https://docs.github.com/en/github/using-git/syncing-a-fork>

Este manual reúne desde conceptos básicos hasta aspectos técnicos precisos para crear, gestionar y publicar proyectos documentales con Quarto, integrando el uso de Git para control de versiones y la publicación eficiente en GitHub Pages.

9 Summary

In summary, this book has no content whatsoever.

References