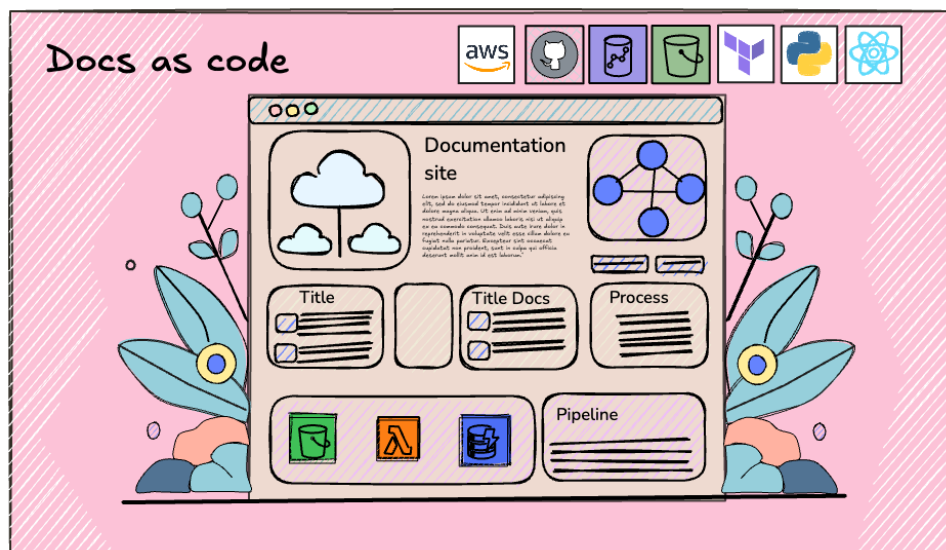


Ilerna

Ciclo Formativo de Grado Superior ASIR

Elementos Técnicos e Implementación del Proyecto MkDocs



Autora: Nerea Fernández

Curso: 2º ASIR

Asignatura: Servicios de Red e Internet

Fecha: 20 de enero de 2026

Índice

1. Introducción	2
2. Extensiones de Markdown	2
2.1. Admonitions (Bloques de Información)	2
2.2. Diagramas con Mermaid	3
2.3. Pestañas de Contenido (Tabbed)	3
2.4. Resaltado de Código y Sintaxis	4
2.5. Tablas de Datos	4
2.6. Iconos y Emojis	4
2.7. Otras Extensiones	5
3. Características y Personalización del Tema Material	5
3.1. Paleta de Colores	5
3.2. Funcionalidades de Navegación y UX	5
3.3. Funcionalidades de Código	6
3.4. Iconos Personalizados	6
4. Plugin de Búsqueda	6
5. Estructura de Navegación del Proyecto	6
6. Resumen de Elementos Utilizados	7

1. Introducción

En el siguiente documento se describe los aspectos técnicos, detalles de diseño y extensiones utilizadas en **MkDocs**. Para crear el sitio estático se ha seleccionado el tema *Material for MkDocs*, y se unió junto con diversas extensiones de Python para mejorar la lectura, la navegación y la presentación de configuraciones técnicas de red.

2. Extensiones de Markdown

2.1. Admonitions (Bloques de Información)

Se utilizó la extensión `admonition` para generar bloques de alertas visuales. Estos elementos son necesarios en documentación técnica para resaltar advertencias o consejos sin interrumpir el flujo de lectura principal.

La configuración utilizada en el archivo `mkdocs.yml` es la siguiente:

```
1 markdown_extensions :
2   - admonition
3   - pymdownx.details
```

Tipo de contenido utilizados en el proyecto:

- **note**: Notas de explicación breve sobre ideas, o comandos.
- **info**: Información extra que sea de interés para el proyecto.
- **tip**: Consejos prácticos o recomendadas.
- **warning**: Advertencias importantes o de riesgo que se pueden evitar.
- **quote**: Anotaciones personales sobre el proyecto realizado.

Ejemplo de implementación en código Markdown:

```
1 !!! tip " Por qu usar VLANs?"
2 Separar la red en VLANs me permite tener m s seguridad,
3 mejor rendimiento y facilita la gesti n.
```

A continuación se muestra como se vería visualmente en el proyecto:

Consejo

¿Por qué usar VLANs?

Separar la red en VLANs me permite tener más seguridad, mejor rendimiento de la red al reducir los dominios de broadcast y facilita la gestión administrativa.

2.2. Diagramas con Mermaid

Para la topología de red se utilizó la integración de diagramas mediante la extensión `pymdownx.superfences`. Esto permite convertir el código de texto en gráficos interactivos.

```
1 markdown_extensions:
2   - pymdownx.superfences:
3       custom_fences:
4         - name: mermaid
5           class: mermaid
6           format: !!python/name:pymdownx.superfences.
              fence_code_format
```

Esta funcionalidad se emplea principalmente en la página de inicio (`index.md`) para mostrar de forma clara:

- La estructura de la topología física en estrella.
- La segmentación lógica de las 4 VLANs con códigos de color distintos.
- Las conexiones clave, entre el Router y el Switch, y los enlaces de acceso a los dispositivos finales.

De esta manera se facilita la comprensión rápida del esquema general del proyecto de red.

2.3. Pestañas de Contenido (Tabbed)

Para organizar la información de manera eficiente y evitar el *scrolling* excesivo en la documentación de redes, se ha utilizado la extensión `pymdownx.tabbed`. Esto permite agrupar la información en un mismo espacio, mejorando la visibilidad.

Las pestañas en `index.md` permiten navegar entre las características de cada segmento de red:

- **VLAN 10** - PCs
- **VLAN 20** - Impresoras
- **VLAN 30** - WiFi (Red de Invitados)
- **VLAN 40** - Router (Gestión)

2.4. Resaltado de Código y Sintaxis

Dado que el proyecto incluye numerosas configuraciones de dispositivos Cisco, el resaltado de sintaxis es necesario para mejorar la experiencia. Se configuró de la siguiente manera:

```
1 markdown_extensions:
2   - pymdownx.highlight:
3       anchor_linenums: true
4       line_spans: __span
5       pygments_lang_class: true
6   - pymdownx.inlinehilite
```

Esto permite que todos los comandos de configuración del switch y router, así como los comandos de verificación (como `show ip interface brief`), aparezcan con una sintaxis coloreada distinguiéndose del resto del texto. Además, se añadieron también comentarios explicativos dentro de los propios bloques de código.

2.5. Tablas de Datos

Se usaron tablas Markdown para estructurar y presentar datos técnicos precisos en varias secciones:

- **Tabla de VLANs** (en `index.md`): Detalla el ID de VLAN, nombre, propósito y rango de red.
- **Tabla de Direccionamiento** (en `configuracion.md`): Especifica las direcciones IP estáticas, máscaras de subred y gateways de cada dispositivo.
- **Tabla de Impresoras**: Listado específico de sus IPs.
- **Tabla de Pruebas** (en `pruebas.md`): Documenta el checklist de las comprobaciones de conectividad realizadas.

Gracias a las tablas la información compleja y detallada esta mejor estructurada y es más sencilla a la hora de consultar.

2.6. Iconos y Emojis

Se utilizó la extensión `pymdownx.emoji` para añadir elementos visuales rápidos, utilizados principalmente en la sección de seguridad para indicar visualmente los permisos de las Listas de Control de Acceso (ACLs):

- **Permitido**: Para indicar que el tráfico está autorizado.
- **Denegado**: Para indicar que el tráfico está bloqueado.

2.7. Otras Extensiones

Se añadieron también las siguientes extensiones:

- `attr_list`: Permite añadir atributos HTML adicionales a los elementos Markdown.
- `md_in_html`: combina sintaxis Markdown dentro de bloques HTML.
- `footnotes`: Para la añadir notas al pie de página.

3. Características y Personalización del Tema Material

3.1. Paleta de Colores

Se configuró un selector de tema (Claro/Oscuro) accesible desde la interfaz, utiliza colores inspirados en el entorno de Cisco Packet Tracer para simular la aplicación:

```
1 palette:
2   - scheme: default # Modo claro
3     primary: blue
4     accent: cyan
5   - scheme: slate # Modo oscuro
6     primary: blue grey
7     accent: cyan
```

3.2. Funcionalidades de Navegación y UX

Para mejorar la experiencia de usuario (UX) y la navegabilidad del sitio, se habilitó las siguientes funcionalidades del tema:

Información

Características de Navegación Activas:

- **navigation.tabs**: Las secciones principales aparecen como pestañas en la barra superior.
- **navigation.sections**: Organiza el contenido en secciones agrupadas en el menú lateral.
- **navigation.top**: Añade un botón flotante para volver al inicio de la página.
- **navigation.footer**: Incluye enlaces de navegación secuencial en el pie de página.
- **search.suggest**: Ofrece sugerencias de búsqueda predictiva mientras se escribe.
- **search.highlight**: Resalta visualmente los términos buscados dentro del contenido.

3.3. Funcionalidades de Código

- **content.code.copy:** Añade un botón de copiado rápido en cada bloque de código.
- **content.code.annotation:** Permite añadir anotaciones interactivas dentro del código.
- **content.tabs.link:** Vincula el estado de las pestañas entre diferentes páginas.

3.4. Iconos Personalizados

Se configuró un icono temático de red como logotipo del sitio para reforzar la identidad del proyecto:

```
1 icon:  
2   logo: material/network  
3   repo: fontawesome/brands/github
```

4. Plugin de Búsqueda

Se ha activado y configurado el plugin de búsqueda nativa. Esto permite buscar el contenido en español:

```
1 plugins:  
2   - search:  
3     lang: es
```

5. Estructura de Navegación del Proyecto

La estructura del proyecto, esta ordenada de manera lógica, permite seguir todo el progreso del proyecto sin problemas:

```
1 nav:  
2   - Inicio: index.md  
3   - Configuraci n de Red: configuracion.md  
4   - Seguridad y ACLs: seguridad.md  
5   - Pruebas y Monitorizaci n: pruebas.md  
6   - Conclusiones: conclusiones.md
```

Este orden permite seguir el proyecto desde la presentación inicial de la topología hasta las conclusiones finales, pasando por las fases de configuración, implementación de seguridad y pruebas de verificación.

6. Resumen de Elementos Utilizados

Categoría	Elementos e Implementación
Motor Base	MkDocs (Generador de sitios estáticos en Python)
Tema Visual	Material for MkDocs (Modos Default y Slate)
Bloques Especiales	Admonitions (note, info, tip, warning, quote)
Diagramas	Mermaid JS para la topología de red
Organización	Pestañas de contenido (Tabbed), Tablas avanzadas
Código	Resaltado de sintaxis (Pygments), Botón de copiar
Visual	Iconos personalizados, Emojis para ACLs
Navegación	Pestañas superiores, Búsqueda predictiva, Footer links
Extensiones Clave	Suite Pymdownx (Superfences, Details, Tabbed, etc.)
Plugins	Search (configurado en español)

Tabla 1: Resumen técnico de los elementos utilizados en el proyecto MkDocs