

# ED06\_2 - Integración de MkDocs con GitHub Pages

## 1. Introducción

En este ejercicio, aprenderás a crear una documentación profesional utilizando MkDocs y a desplegarla en GitHub Pages. MkDocs es una herramienta que te permite generar sitios web de documentación a partir de archivos Markdown. Por otro lado, GitHub Pages es un servicio de GitHub que te permite alojar sitios web estáticos de manera gratuita.

A continuación se detallan todos los pasos a seguir para crear tu documentación con MkDocs y desplegarla en GitHub Pages.

## 2. Creación de Documentación con MkDocs y GitHub Pages

### 2.1. Prerrequisitos

- Tener Git instalado en tu sistema
- Tener Python instalado (versión 3.7 o superior)
- Tener una cuenta en GitHub
- Tener conocimientos básicos de Markdown

### 2.2. Paso 1: Configuración del Entorno

1. Primero, crea un nuevo directorio para tu proyecto:

```
mkdir mi-documentacion
cd mi-documentacion
```

2. Crea un entorno virtual de Python: un entorno virtual es una herramienta que te permite aislar las dependencias de tu proyecto para evitar conflictos con otras aplicaciones.

```
python -m venv venv
```

### 3. Activa el entorno virtual:

- En Windows:

```
venv\Scripts\activate
```

- En Linux/Mac:

```
source venv/bin/activate
```

### 4. Instala MkDocs y el tema Material:

```
pip install mkdocs mkdocs-material
```

Puedes realizar todos estos pasos desde la consola de comandos de Visual Studio Code, sobre el directorio de tu proyecto.

## 2.3. Paso 2: Crear el Proyecto MkDocs

#### 1. Inicializa un nuevo proyecto MkDocs en el directorio actual:

```
mkdocs new .
```

#### 2. Esto creará dos archivos importantes:

- `mkdocs.yml` : El archivo de configuración
- `docs/index.md` : Tu página principal

## 2.4. Paso 3: Configurar MkDocs

#### 1. Abre el archivo `mkdocs.yml` y configura el tema Material:

```
site_name: Mi Documentación
theme:
  name: material
  features:
    - navigation.tabs
    - navigation.sections
    - navigation.expand
    - content.code.copy
```

## 2.5. Paso 4: Crear la Estructura de Documentación

1. En la carpeta `docs`, crea la siguiente estructura:

```
docs/
├── index.md
├── guia/
│   ├── instalacion.md
│   ├── configuracion.md
│   └── despliegue.md
└── ejemplos/
    └── ejemplo1.md
```

La estructura de carpetas y archivos es opcional y puedes organizar tu documentación de la manera que prefieras. Se proporciona un ejemplo para que tengas una idea de cómo estructurar tu documentación. El contenido de los archivos es simplemente ilustrativo.

2. Edita el archivo `docs/index.md` (este paso es opcional si utilizas la estructura de carpetas y archivos proporcionada):

```
# Bienvenido a Mi Documentación
```

Esta es la página principal de mi documentación.

```
## Contenido
```

```
- [Guía de Instalación](guia/instalacion.md)
- [Configuración](guia/configuracion.md)
- [Despliegue](guia/despliegue.md)
```

```
...
```

Puedes probar a añadir a esta página elementos como los vistos en el apartado 6.3.

## 2.6. Paso 5: Crear un Repositorio en GitHub

1. Ve a GitHub y crea un nuevo repositorio. Asegurate de crearlo como un **repositorio público** para que la documentación sea accesible en GitHub Pages.
2. Inicializa Git en tu proyecto local:

```
git init
```

3. Crea un archivo `.gitignore` en el directorio raíz de tu proyecto con el siguiente contenido:

venv/

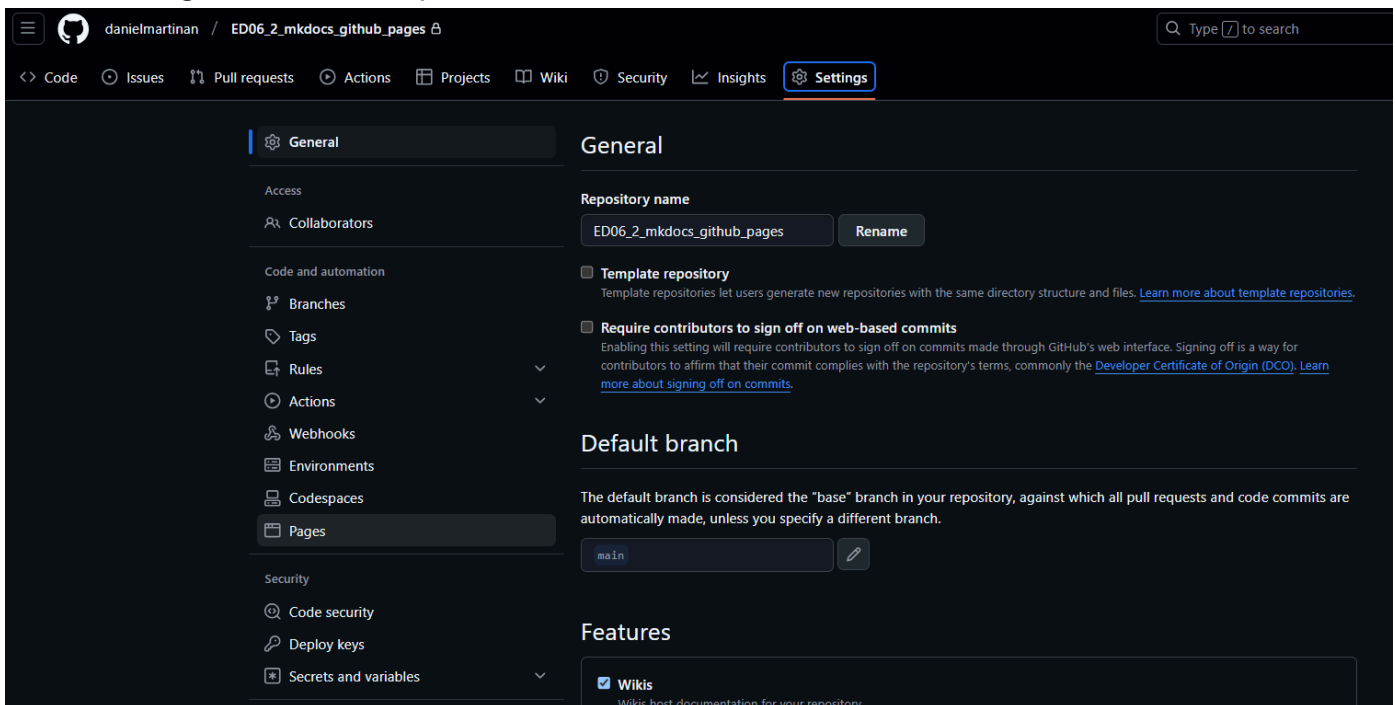
4. Realiza el primer commit y vincula tu repositorio remoto:

```
git add .
git commit -m "Primer commit"
git branch -M main
git remote add origin https://github.com/tu-usuario/tu-repositorio.git
git push -u origin main
```

Asegúrate de reemplazar `tu-usuario` y `tu-repositorio` con tu nombre de usuario y el nombre de tu repositorio en GitHub. Puedes realizar todos estos pasos desde las facilidades proporcionadas por Visual Studio Code, si lo prefieres.

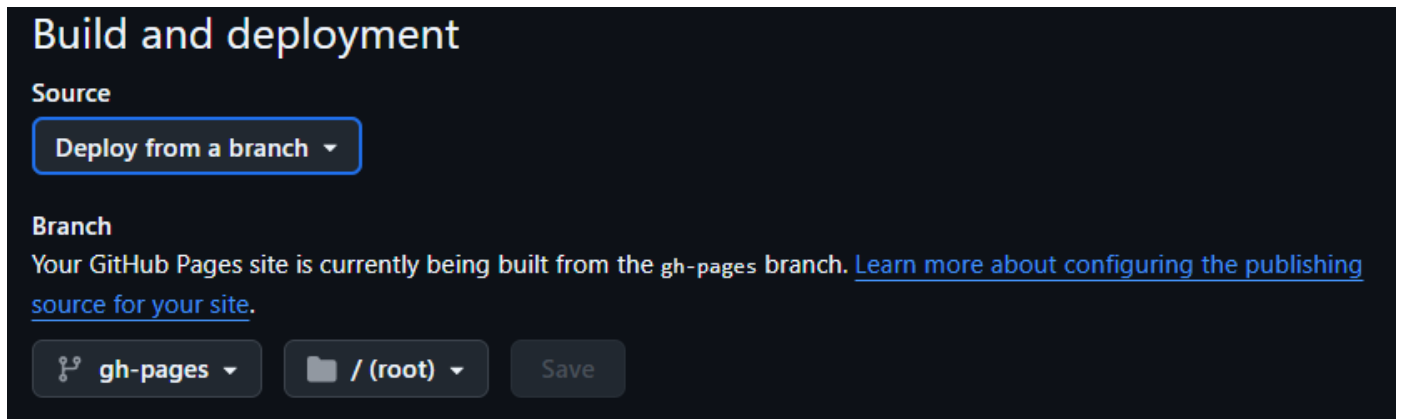
## 2.7. Paso 6: Configurar GitHub Pages

1. Ve a la configuración de tu repositorio en GitHub



2. Navega a la sección "Pages"

3. En "Source", selecciona `Deploy from a branch`, y elige `gh-pages` como rama. Guarda los cambios. Si no te aparece la opción de `gh-pages`, es posible que necesites desplegar la documentación al menos una vez para que aparezca. Continúa con el proceso y retoma este paso más tarde (se indica en el paso 8).



4. Crea un nuevo archivo `.github/workflows/deploy.yml` en el directorio raíz de tu repositorio con el siguiente contenido:

```
name: Deploy MkDocs
on:
  push:
    branches: [ main ]

jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - uses: actions/setup-python@v2
        with:
          python-version: '3.x'
      - run: pip install mkdocs-material
      - run: mkdocs gh-deploy --force
```

Con este archivo, se configurará un flujo de trabajo de **GitHub Actions** que desplegará tu documentación en GitHub Pages cada vez que hagas un push a la rama `main`. Puedes encontrar más información sobre GitHub Actions en la [documentación oficial](#).

## 2.8. Paso 7: Configurar Github actions

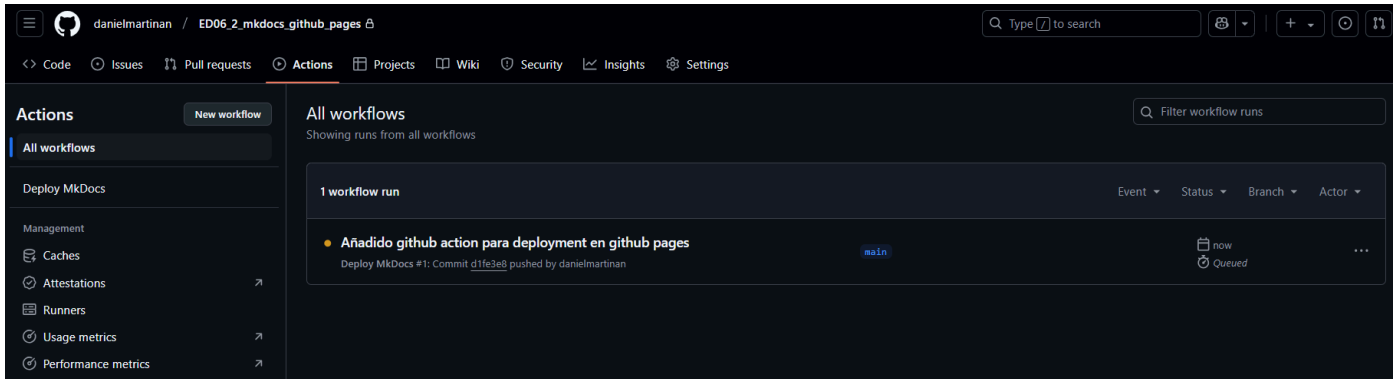
1. Ve a la sección "Actions" de tu repositorio, y selecciona la sección General.
2. En la sección Workflow Permissions, selecciona "Read and write access" para permitir que el flujo de trabajo pueda hacer push a la rama `gh-pages`.

## 2.9. Paso 8: Desplegar la Documentación

1. Haz commit de los cambios en el archivo `.github/workflows/deploy.yml` y realiza un push a la rama `main`:

```
git add .
git commit -m "Configuración de GitHub Pages"
git push origin main
```

2. Ve a la sección "Actions" de tu repositorio para ver el progreso del despliegue



3. Una vez completado, tu documentación estará disponible en:  
<https://tu-usuario.github.io/tu-repositorio>

Es posible que surja algún error en el despliegue si no se han seguido correctamente los pasos anteriores. En ese caso, revisa los mensajes de error en la sección "Actions" de tu repositorio para identificar el problema.

The screenshot shows the GitHub Actions interface for a repository named 'ED06\_2\_mkdocs\_github\_pages'. The workflow is titled 'Deploy MkDocs' and is currently in a failed state, indicated by a red 'X' icon and the text 'Añadido github action para deployment en github pages #1'. The workflow was triggered via a push 2 minutes ago by 'danielmartinan' on the 'main' branch. The status is 'Failure', the total duration is '21s', and there are no artifacts. The workflow file is 'deploy.yml', which is triggered on 'push'. The workflow has one job named 'deploy', which failed with an exit code of 1. The error message is 'Process completed with exit code 1.'.

← Deploy MkDocs

✖ Añadido github action para deployment en github pages #1

Summary

Jobs

✖ deploy

Run details

Usage

Workflow file

Triggered via push 2 minutes ago

Status

Total duration

Artifacts

danielmartinan pushed -> d1fe3e8 main

Failure

21s

—

deploy.yml

on: push

✖ deploy 10s

Annotations

1 error

✖ deploy

Process completed with exit code 1.

Si en el paso 6 no conseguiste configurar GitHub Pages correctamente, vuelve a intentarlo después de haber desplegado la documentación al menos una vez. Configura `source` de GitHub Pages a `Deploy from a branch`, selecciona `gh-pages` y vuelve a intentar el despliegue.

El despliegue puede demorarse unos segundos, así que comprueba que el despliegue se ha realizado correctamente después de 30 o 60 segundos.

Puedes ver cómo se ha desplegado la documentación de este tutorial en [este enlace](#).

## 3. Recursos Útiles

- [Documentación oficial de MkDocs](#)
- [Documentación del tema Material](#)
- [GitHub Pages](#)

## 4. Entrega

Una vez realizados todos los pasos, debes entregar un archivo PDF con las capturas de pantalla que muestren el proceso de creación de la documentación con MkDocs y su despliegue en GitHub Pages.

Asegúrate de incluir capturas de pantalla de la configuración de GitHub Pages y del flujo de trabajo de GitHub Actions.

Además, debes añadir un enlace a tu repositorio de GitHub donde se encuentre alojada la documentación así como a la documentación desplegada en GitHub Pages.

Añade todos los comentarios que consideres necesarios para explicar el proceso seguido, así como las decisiones tomadas a lo largo del proceso.

## **5. Criterios de Evaluación**

Esta práctica se evaluará como apto o no apto, según se consiga desplegar la documentación o no.