

# DocFX Unity Documentation Generator

A simple tool to generate API documentation for Unity projects using DocFX.

## Features

- Generates API documentation from C# code
- Creates PDF documentation
- Provides local preview server
- Supports GitHub Pages deployment
- Includes .gitignore rules for local builds
- Unity-specific documentation templates and styles

## Quick Start

1. Install DocFX using Chocolatey:

```
choco install docfx
```

Or follow the installation guide at <https://dotnet.github.io/docfx/index.html>  
(<https://dotnet.github.io/docfx/index.html>)

2. Copy the `Documentation` folder and `build.ps1` to your project's root directory
3. Configure your documentation:

- a. Update `Documentation/docfx.json`:

```
{
  "sitemap": {
    "baseUrl": "https://yourusername.github.io/YourRepository/"
  },
  "globalMetadata": {
    "_appName": "Your Project Documentation",
    "_appTitle": "Your Project Documentation",
    "_enableSearch": true
  }
}
```

- b. Modify `Documentation/filterConfig.yml`:

```
apiRules:
- include:
  uidRegex: ^YourNamespace
  type: Namespace
```

Replace `YourNamespace` with your project's root namespace.

4. Add the following rules to your `.gitignore` file:

```
# Ignore local DocFX generated files
Documentation/_site/ # Generated website files
Documentation/api/   # Generated API documentation
Documentation/obj/   # DocFX temporary files
```

These rules prevent locally generated documentation from being committed to your repository. This is important because:

5. Documentation should be generated by GitHub Actions for deployment
6. Prevents repository bloat
7. Avoids conflicts between local and CI-generated files
8. Generate documentation by running:

```
powershell -ExecutionPolicy Bypass -File build.ps1
```

This will:

- Generate API documentation
- Create PDF files
- Start a local preview server at <http://localhost:8080> (<http://localhost:8080>)

**Note:** Due to .gitignore rules, the generated files will be available locally but won't be committed to the repository. The official documentation will be generated and deployed by GitHub Actions.

## Unity-Specific Setup

This generator is specifically configured for Unity projects and includes:

- Unity-friendly documentation templates
- Proper handling of Unity-specific types and namespaces
- Integration with Unity's XML documentation format

## GitHub Pages Setup

### 1. Setup GitHub Actions:

- Copy `.github/workflows/documentation.yml` from this repository to your project's `.github/workflows/` directory
- If the `.github/workflows` directory doesn't exist, create it first
- Commit and push the changes

### 2. Configure GitHub Pages:

- Go to your repository's Settings
- Navigate to Pages section
- Set the following:
  - Source: Deploy from a branch
  - Branch: gh-pages
  - Folder: / (root)
- Wait for GitHub Actions to complete the deployment
- Your documentation will be available at <https://yourusername.github.io/YourRepository/>

# Local Development Structure

Basic structure for local development and testing:

```
YourProject/
├── Documentation/           # Documentation source files
│   ├── api/                # API documentation (generated, git-ignored)
│   ├── _site/              # Generated website (git-ignored)
│   ├── _pdf/               # Generated PDFs (git-ignored)
│   ├── docfx.json          # DocFX configuration
│   ├── filterConfig.yml    # API filter rules
│   ├── toc.yml             # Table of contents
│   └── index.md            # Documentation home page
├── .gitignore              # Git ignore rules
└── build.ps1               # Build script
```

This is the minimal structure needed for local documentation generation and preview.

## Complete Project Structure with GitHub Pages

When setting up GitHub Pages deployment, your project structure will expand to include GitHub Actions workflow:

```
YourProject/
├── .github/
│   └── workflows/          # GitHub Actions workflows
│       └── documentation.yml # Documentation generation workflow
├── Documentation/         # Documentation source files
│   ├── api/               # API documentation (generated, git-ignored)
│   ├── _site/              # Generated website (git-ignored)
│   ├── _pdf/               # Generated PDFs (git-ignored)
│   ├── styles/             # Styles for pdf generation
│   ├── templates           # Design template
│   ├── docfx.json          # DocFX configuration
│   ├── filterConfig.yml    # API filter rules
│   ├── toc.yml             # Table of contents
│   └── index.md            # Documentation home page
├── .gitignore              # Git ignore rules
└── build.ps1               # Build script
```

This structure includes everything needed for both local development and automated deployment.

## File Management

- **Source Files:** Templates, configuration files, and markdown content are tracked in git
- **Generated Files:** The following are generated locally but ignored by git:
  - Documentation/\_site/ : Generated website files
  - Documentation/api/ : API documentation
  - Documentation/obj/ : DocFX cache and temporary files
- **GitHub Pages:** Documentation is automatically generated and deployed by GitHub Actions

## Troubleshooting

Common issues and solutions:

### 1. PDF Generation Fails

- Make sure wkhtmltopdf is installed
- Check if your documentation contains valid markdown

### 2. API Documentation Missing

- Verify your namespace in filterConfig.yml
- Ensure your C# files have proper XML documentation comments

### 3. GitHub Pages Not Updating

- Check GitHub Actions workflow runs
- Verify gh-pages branch exists and is configured correctly

### 4. Unity Scripts Not Appearing

- Make sure scripts have proper XML documentation comments
- Check if namespace matches filterConfig.yml settings
- Verify script files are included in the build

## License

This project is licensed under the MIT License - see the [LICENSE \(LICENSE\)](#) file for details.

## Contributing

Contributions are welcome! Feel free to submit issues and pull requests.

## Acknowledgments

- [DocFX \(https://dotnet.github.io/docfx/\)](https://dotnet.github.io/docfx/) for the documentation generation engine
- [Unity \(https://unity.com/\)](https://unity.com/) for the game engine
- [DocFxForUnity \(https://github.com/NormandErwan/DocFxForUnity\)](https://github.com/NormandErwan/DocFxForUnity) - guide for DocFX integration with Unity
- [UnityFX \(https://github.com/code-beans/UnityFX\)](https://github.com/code-beans/UnityFX) for Unity documentation style and templates