

# Sorbieskis Documentation

---

## Complete User Manual

*Sorbieskis*

*Copyright © 2024 Sorbieskis*

Table of contents

---

1. Welcome to Sorbieskis Docs Demo	3
1.1 How It Works	3
1.2 Features	3
1.3 Getting Started	3
1.4 Technical Architecture	4
1.5 Sample Content	4
1.6 Next Steps	5
1.7 🎉 LIVE EDITING DEMONSTRATION	5
2. Live Editing Test Page	6
2.1 Real-Time Features	6
2.2 ✅ Live Edit Success!	6

# 1. Welcome to Sorbieskis Docs Demo

---

This is a demonstration of a complete **docs-as-code** system that combines:

- **Web-based editing** through Decap CMS
- **Static site generation** with MkDocs Material theme
- **Professional PDF generation** using LaTeX templates
- **Automated deployment** via GitHub Actions

## 1.1 How It Works

---

```
graph LR
  A[Edit via Web CMS] --> B[Commit to GitHub]
  B --> C[GitHub Actions Trigger]
  C --> D[Build Website]
  C --> E[Generate PDF]
  D --> F[Deploy to GitHub Pages]
  E --> G[Upload PDF Artifact]
```

## 1.2 Features

---

### 1.2.1 Web Editing Interface

---

Edit documentation directly in your browser using the intuitive Decap CMS interface:

- Rich text editor with live preview
- GitHub integration for version control
- No technical setup required for content editors

### 1.2.2 Professional Output

---

Generate both web and PDF versions from the same source:

- **Website:** Modern, responsive design with Material theme
- **PDF:** Professional typography with LaTeX quality
- **Consistency:** Same content, multiple formats

### 1.2.3 Automated Workflow

---

Every edit triggers automatic updates:

1. Content is saved via the web interface
2. GitHub Actions builds both website and PDF
3. Website deploys to GitHub Pages
4. PDF is available as a download artifact

## 1.3 Getting Started

---

### 1.3.1 For Content Editors

---

1. Navigate to `/admin/` on the deployed site
2. Authenticate with GitHub

3. Start editing content immediately

### 1.3.2 For Developers

- 1. Clone this repository
- 2. Install dependencies: `pip install -r requirements.txt`
- 3. Run locally: `mkdocs serve`
- 4. Access CMS at `http://localhost:8000/admin/`

## 1.4 Technical Architecture

This system demonstrates modern documentation practices:

- **Single Source of Truth:** All content in Markdown format
- **Version Control:** Full history and collaboration via Git
- **Automated Testing:** CI/CD pipeline validates all changes
- **Multiple Outputs:** Web and PDF from same source files
- **Professional Quality:** Typography and design suitable for enterprise use

## 1.5 Sample Content

### 1.5.1 Code Examples

```
def generate_docs():
    """
    Automatically generate documentation from Markdown source.
    """
    build_website()
    generate_pdf()
    deploy_to_production()
```

### 1.5.2 Admonitions



**Documentation Note**

This content is automatically synchronized between the website and PDF versions.



**Important**

Always test changes in a preview environment before publishing.



**Pro Tip**

Use the web editor for quick changes, but consider using a local environment for major restructuring.

1.5.3 Tables

Feature	Website	PDF
Search	✓	✗
Navigation	✓	✓
Typography	✓	✓
Images	✓	✓
Links	✓	✓

1.6 Next Steps

This demo provides a foundation for:

- Enterprise documentation systems
- Technical writing workflows
- Collaborative content creation
- Multi-format publishing pipelines

Customize the theme, add more content, and configure the system for your specific needs.

Generated automatically from Markdown source. Last updated: Sorbieskis Docs Demo build system.

1.7 🎨 LIVE EDITING DEMONSTRATION

**This content was just added via live editing!**

If you're viewing this in a browser at `http://localhost:8000`, you should see this content appear automatically without refreshing the page.

1.7.1 ✓ What's Working:

- **Live Reload:** ✓ Edit files → Instant browser refresh
- **Professional Theme:** ✓ Material Design + syntax highlighting
- **Multiple Formats:** ✓ Website + PDF from same source
- **Search:** ✓ Full-text search with auto-indexing
- **Navigation:** ✓ Auto-generated from content structure

1.7.2 🛠 CMS Status:

- **Interface:** ✓ Available at `/admin/`
- **Authentication:** ⚠ Requires GitHub OAuth setup
- **File Editing:** ✓ Direct file editing works perfectly

**Try editing this file ( docs/index.md ) and watch it update instantly!**

## 2. Live Editing Test Page

**Created at:** 2024-07-30 14:55

This page demonstrates the live editing capabilities of the docs-as-code system.

### 2.1 Real-Time Features





#### 2.1.1 Instant Reload

- Edit this file and save
- Watch the browser auto-refresh
- Changes appear immediately

#### 2.1.2 Rich Content Support

```
def live_editing_demo():  
    """Demonstrate code highlighting in live preview."""  
    print("Content updates instantly!")  
    return "Live editing works!"
```

#### 2.1.3 Data Visualization

Feature	Status	Notes
Live Reload	 Active	Changes visible instantly
Syntax Highlighting	 Working	Code blocks render properly
Math Support	 Ready	$E = mc^2$
Admonitions	 Styled	See below

#### Live Editing Tip

Try editing this content and watch it update in real-time in your browser!

#### 2.1.4 Testing Instructions

1. **Open browser:** <http://127.0.0.1:8000/docs-as-code-demo/test-live-editing/>
2. **Edit this file:** Change any content above
3. **Save the file:** Ctrl+S or :w in vim
4. **Watch browser:** Should auto-refresh with changes

Last updated: Successfully demonstrated live editing at \$(date)!

### 2.2 Live Edit Success!

This content was just added via live editing demonstration. The browser should automatically refresh to show this new content!