

# Author Quick Start — Simple-CMS

## Prototype

One-page guide for authors to create, validate, collaborate, and publish lessons.

1) Open repository in VS Code - File → Open Folder → select repository root. - Accept recommended extensions (YAML, Markdown, markdownlint, Live Share).

2) Scaffold a lesson (recommended)

```
# scaffold and open in VS Code
python .\scripts\create_lesson.py "My Lesson Title" --course Microsoft
```

Or copy the template manually:

```
mkdir -Force Content\Microsoft-Security\Module-01
Copy-Item Templates\lesson-template.md Content\Microsoft-Security\Mo
code Content\Microsoft-Security\Module-01\my-lesson-001.md
```

3) Edit frontmatter and content - Update YAML frontmatter fields: `id`, `title`, `course`, `module`, `status` (Draft | Review | Approved | Published), `authors`. - Save frequently. Use Live Share for pair editing.

4) Validate metadata and sync index

```
python .\scripts\check_status_consistency.py --fix
```

5) When ready for review - Update frontmatter to `status: "Review"` and push a branch or invite reviewers via Live Share.

6) Approve and publish

When a reviewer decides the lesson is final, run:

```
python .\scripts\workflow_transition.py <article-id> Approved
```

This will move the file into `Content/Published/` and update `content_status.json` to Published automatically.

7) Commit & open PR

```
git add Content\<course>\<module>\*.md content_status.json
git commit -m "Add/update <article-id>"
git push --set-upstream origin demo/<your-branch>
# Open PR via GitHub web UI or use `gh pr create` (if you have GitHub CLI)
```

Notes - Use branches for parallel work or Live Share for real-time collaboration. - If the checker reports parse errors, install `pyyaml` in the environment used by `python`:

```
python -m pip install pyyaml
```

If you want, I can create and open the PR for you (I need `gh` auth or a GitHub token), or you can click the PR creation link that I've opened in your browser.