

Perfekt 🙌 — hier ist Ihre **Mini-Anleitung für das Blogsystem „Errata & Ergänzungen“**

auf der *Pilz-Chronik Bonusseite* – einfach, klar und exakt auf Ihr Setup abgestimmt.

Ziel

Sie können künftig jede kleine Änderung, Richtigstellung oder neue Quelle in Ihrer *Pilz-Chronik* als **Blogbeitrag** hinzufügen — ohne Layout- oder Technikaufwand.

GitHub Pages zeigt die Beiträge automatisch chronologisch auf Ihrer Startseite an.

1. Wo die Beiträge liegen

Alle Einträge kommen in den Ordner:

`_posts/`

Jeder Beitrag ist **eine Markdown-Datei (.md)**.

Beispiel:

`_posts/2025-10-25-neue-daten-zum-zweig-reider.md`

2. Dateiname

Der Dateiname folgt immer dem Schema:

YYYY-MM-DD-titel-des-beitrags.md

- YYYY = Jahr (vierstellig)
- MM = Monat (zweistellig)
- DD = Tag (zweistellig)
- titel-des-beitrags = kurze, sprechende Bezeichnung (nur Kleinbuchstaben und Bindestriche)



Beispiel:

2025-10-25-ergaenzung-zum-eberstaller-tagebuch.md



3. Aufbau eines Beitrags

Jeder Beitrag beginnt mit dem sogenannten *Front Matter*
(das sind die Metadaten zwischen den drei Bindestrichen):

```
---  
layout: post  
title: "Ergänzung: Neue Tagebuchseite entdeckt"  
date: 2025-10-25  
categories: errata  
---
```

Bei der Sichtung der Digitalisate im Archiv
Leitmeritz wurde eine weitere Seite
aus dem Tagebuch von Johann Eberstaller gefunden
(Eintrag vom 11.05.1674).
Der Text bestätigt die Erwähnung der Familie Reider
während des Kriegsjahres 1672.

Erklärung:

Feld Bedeutung

layout: post Steuert die Anzeige als Blogeintrag

title: Überschrift, erscheint automatisch auf der Startseite

date: Veröffentlichungsdatum

categories: dient zur Gruppierung (bei Ihnen z. B. errata oder ergaenzungen)

Inhalt Ihr eigentlicher Text (Markdown-Format möglich)

4. Empfohlene Textlänge

- Ein kurzer Absatz reicht völlig – 5–10 Zeilen sind optimal.
- Für längere Beiträge (z. B. mehrere Quellenvergleiche) können Sie Zwischenüberschriften nutzen:


Neue Quellen

Bewertung

5. Veröffentlichung

1. Öffnen Sie Ihr GitHub-Repository `pilzchronik.github.io/bonusseite/`.
2. Navigieren Sie zu `_posts/`.
3. Klicken Sie auf **“Add file”** → **“Create new file”**.
4. Geben Sie den neuen Dateinamen ein (z. B. `2025-10-25-neuer-eintrag.md`).

5. Fügen Sie den Markdown-Text ein.
6. Klicken Sie unten auf **“Commit new file”**.

 Nach 1–2 Minuten erscheint der Beitrag automatisch auf der Startseite – ganz oben.

6. Optional: Kategorienübersicht

Wenn Sie viele Beiträge haben, können Sie später Kategorien auswerten, z. B.:

- `/bonusseite/category/errata/`
- `/bonusseite/category/ergaenzungen/`

Das lässt sich später einfach in `_config.yml` aktivieren, wenn Sie möchten.

7. Stil-Tipp

Bleiben Sie beim sachlich-historischen Stil, wie in Ihrer Chronik:

- Keine Anrede, keine persönliche Form („ich habe gefunden“)
 - Lieber: „Bei FamilySearch wurde ein weiterer Eintrag entdeckt.“
 - Einheitlich mit österreichischem Deutsch („Jänner“, „bis dato“, „Kassa“ etc.)
-

Fazit

Vorteil Beschreibung



Einfach

Nur eine Markdown-Datei pro Meldung



Chronologisch

Sortiert automatisch nach Datum



Dokumentiert

Ideal für Nachträge, Korrekturen und Quellenhinweise



Dauerhaft

Beiträge bleiben archiviert und zitierfähig

Möchten Sie, dass ich Ihnen jetzt noch **eine druckfreundliche PDF-Version dieser Mini-Anleitung** (1 Seite) erstelle,
die Sie als Referenz beim Arbeiten mit GitHub behalten können?