Didattica libera

Breve guida a LaTeX

Luca Baldini

https://taleae.codeberg.page

Versione 0.0.0 (cronologia)

Prefazione

Questa è una breve guida per pubblicare su Codeberg Pages libri realizzati con LaTeX a partire dal template latex_template.

Buon divertimento!

Licenza

Tutto il materiale dell'opera è distribuito con licenza *Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).*



© 2026 Luca Baldini

Tu sei libero di:

Condividere—riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare questo materiale con qualsiasi mezzo e formato per qualsiasi fine, anche commerciale.

Modificare—remixare, trasformare il materiale e basarti su di esso per le tue opere per qualsiasi fine, anche commerciale.

Il licenziante non può revocare questi diritti fintanto che tu rispetti i termini della licenza.

Alle seguenti condizioni:

Attribuzione—Devi riconoscere una menzione di paternità adeguata, fornire un link alla licenza e indicare se sono state effettuate delle modifiche. Puoi fare ciò in qualsiasi maniera ragionevole possibile, ma non con modalità tali da suggerire che il licenziante avalli te o il tuo utilizzo del materiale.

StessaLicenza—Se remixi, trasformi il materiale o ti basi su di esso, devi distribuire i tuoi contributi con la stessa licenza del materiale originario.

Divieto di restrizioni aggiuntive—Non puoi applicare termini legali o misure tecnologiche che impongano ad altri soggetti dei vincoli giuridici su quanto la licenza consente loro di fare.

Note

Non sei tenuto a rispettare i termini della licenza per quelle componenti del materiale che siano in pubblico dominio o nei casi in cui il tuo utilizzo sia consentito da una eccezione o limitazione prevista dalla legge.

Non sono fornite garanzie. La licenza può non conferirti tutte le autorizzazioni necessarie per l'utilizzo che ti prefiggi. Ad esempio, diritti di terzi come i diritti all'immagine, alla riservatezza e i diritti morali potrebbero restringere gli usi che ti prefiggi sul materiale.

Contents

Contents **7**

1. Struttura del repositorio **9**
1.1. Descrizione dei file 9
1.2. Testo e figure 9
1.3. Controllo delle versioni 9

2. Iniziare un nuovo progetto **11**
2.1. Creazione del repositorio 11
2.2. Modifiche iniziali 11
2.3. Scelta della lingua 11
2.4. Scelta della licenza 12
2.5. C'è altro? 12

3. Composizione **13**

4. Rilascio **15**
4.1. Rilascio manuale 15
4.1.1. Configurazione iniziale 15
4.1.2. Workflow di rilascio 15

A. Cronologia delle versioni **17**

1. Struttura del repositorio

Questo capitolo vuole essere una breve guida alla struttura del repositorio ed alle funzionalità che esso offre.

1.1. Descrizione dei file

Da un punto di vista del documento propriamente detto, i file principali sono:

- `main.tex`: contiene la struttura di alto livello del documento, inclusi i metadati rilevanti, le impostazioni di quarto, e l'indice dei file;
- `meta.tex`: è un file ausiliario che contiene la versione corrente del documento;
- `references.bib`: è il file di bibliografia, che nella versione iniziale contiene una sola entry di esempio.

Ci sono poi alcuni file ancillari:

- `README.md`: è il README di più alto livello del repositorio—quello che appare nella pagina principale del repositorio su Codeberg;
- `LICENSE`: è banalmente il file della licenza, che dovrebbe essere coerente con la sezione corrispondente del libro;

Ultimo, ma non per importanza, il flusso di lavoro:

- `Makefile`: contiene i comandi della shell per le operazioni principali—compilazione e rilascio.

1.2. Testo e figure

La struttura del repositorio è pensata in modo che i capitoli, e eventualmente le appendici, siano file `.tex` separati nella cartella `chapters`, mentre le figure siano fisicamente nella cartella `figures`.

La versione iniziale del repositorio ha una manciata di figure nella sotto-cartella `figures/static` con l'idea che eventuali figure generate programmaticamente potrebbero essere collocate in una nuova cartella `figures/generated`, mentre gli script per generarle potrebbero andare in `figures/scripts`.

Ovviamente questa struttura di base può essere modificata a piacimento.

1.3. Controllo delle versioni

Per comodità il numero di versione è contenuto nel file separato `meta.tex`.

Questo file può essere editato manualmente per tracciare la versione del documento ad ogni rilascio.

2. Iniziare un nuovo progetto

Questo capitolo è una breve guida a come creare da zero un nuovo testo da ospitare su Codeberg utilizzando LaTeX.

2.1. Creazione del repositorio

Quanto crei un nuovo repositorio nel tuo spazio utente di Codeberg, hai la scelta di partire da un template: seleziona `taleae/quarto_template`, spuntando *Git content (default branch)* tra i *Template items*.

A questo punto puoi clonare il repositorio localmente ed iniziare a lavorare nella tua *working copy*, e.g., per questa guida

```
git clone ssh://git@codeberg.org/taleae/latex_guide.git
cd latex_guide
git status
```

(Ovviamente devi modificare opportunamente l'url in modo da puntare al tuo nuovo repositorio appena creato.)

Va da sé: è necessario avere LaTeX installato sul tuo computer.

2.2. Modifiche iniziali

Il template su Codeberg è costruito in modo da minimizzare il numero di cambiamenti necessari per cominciare a lavorare sul materiale, ma non fa magie. (Non ancora, per lo meno.)

Ci sono alcune modifiche da fare prima di cominciare, ed in particolare:

- nel file `main.tex`: devi aggiornare opportunamente i campi `${TITLE}` e `${AUTHORS}`;
- nel file `chapters/cc-by-sa-it.tex` (o `chapters/cc-by-sa-en.text`, a seconda della lingua) devi aggiornare il campo `${AUTHORS}`.

2.3. Scelta della lingua

Se il progetto è in italiano non c'è molto altro da fare, a parte rimuovere dal repositorio il file di licenza in inglese (che in questa configurazione non è usato) se proprio tieni all'economia concettuale.

Se, viceversa, il progetto è in inglese, allora devi modificare ulteriormente il file `main.tex`, cambiando in `en` tutte le istanze di `it`. Dovrebbe essere semplice. (Ovviamente dovrai tradurre anche il poco testo già presente dall'italiano all'inglese.)

2.4. Scelta della licenza

Se la licenza di default CC BY-SA 4.0 va bene non devi fare niente.

In caso contrario dovrai modificare opportunamente il file `LICENSE` e creare un nuovo file `.tex` per rimpiazzare il vecchio `chapters/cc-by-sa-it.tex`. (Il riferimento al vecchio file in `main.tex` andrà modificato opportunamente.)

2.5. C'è altro?

Non direi. Siamo pronti per cominciare a scrivere il materiale.

3. Composizione

The Not So Short Introduction to LaTeX2e è una delle guide più autorevoli per cominciare a scrivere in LaTeX.

Il flusso di lavoro tipico consiste nell'aggiungere via via i file dei nuovi capitoli nella cartella `chapters`, aggiornare l'indice nel file `main.tex` e dare libero sfogo alla propria creatività.

Il Makefile del repositorio dovrebbe illustrate in modo auto-esplicativo le operazioni di base rilevanti per il lavoro giornaliero:

- `pdf`: compilazione del documento in pdf;
- `clean` e `cleanall`: rimozione degli artefatti;
- `publish`: rilascio su Codeberg Pages.

4. Rilascio

Ci sono due possibilità per sincronizzare il repositorio con gli artifatti (i.e., il file pdf) visibili pubblicamente da web: automaticamente, attraverso una *action* di Forgejo, oppure manualmente.

Nello spirito di massima semplicità, consigliamo la seconda opzione.

4.1. Rilascio manuale

Il modo più semplice per rendere visibili gli artefatti su Codeberg Pages è pubblicare il contenuto della cartella `_book` in un branch di git chiamato *pages*, ed aggiungere un apposito *webhook* al repositorio che punti a questo branch.

4.1.1. Configurazione iniziale

Sulla pagina web del repositorio:

- vai su "Settings" (la piccola icona di una chiave inglese in alto a destra);
- selezione "Webhooks" nella barra verticale di navigazione a sinistra;
- premi il bottone "Add webhook" e selezione "Forgejo";
- imposta lo url di destinazione come `https://${USER_NAME}.codeberg.page/${REPO_NAME}/` (sostituisci il nome utente ed il nome del repositorio opportuni);
- imposta il campo "Branch filter" a 'pages'
- salva premendo il bottone "Add webhook".

Tutto questo va fatto solo la prima volta.

4.1.2. Workflow di rilascio

Il modo più comodo per effettuare il rilascio è utilizzare il tool ghp-import. Questa piccola utility è pensata essenzialmente per copiare il contenuto di una cartella—in questo caso il pdf in output da LaTeX—in un branch di git. Esattamente quello che ci serve.

Se non hai già 'ghp-import' sulla tua macchina dovrai installarlo la prima volta. Come tutti i pacchetti Python, si fa banalmente con pip:

```
python -m pip install ghp-import
```

Il flusso di lavoro tipico per un rilascio manuale è

```
make pdf
ghp-import -o -f -p -b pages _book
```

ovverosia:

1. rigeneriamo completamente il pdf in uscita;
2. copiamo il contenuto sul branch `pages` e facciamo il push dei cambiamenti sul repositorio remoto.

Per la precisione, il significato delle opzioni di ‘ghp-import‘ è più o meno

- ‘-o‘: forza un nuovo commit senza la history precedente;
- ‘-f‘: forza il push al repositorio;
- ‘-p‘: fai il push sul repositorio remoto dopo il commit;
- ‘-b pages‘: specifica il nome del branch di destinazione.

Attenzione: `ghp-import` rigenera completamente ogni volta il branch di destinazione, sostituendo l’intero contenuto e riscrivendo la history. Per questo particolare caso d’uso è esattamente quello che vogliamo, ma ricordate che di non editare manualmente il branch `pages`, perché qualsiasi cambiamento andrà perduto al prossimo rilascio.

Se questo sembra complicato, la buona notizia è che il Makefile nel repositorio template ha un target `publish` che fa esattamente questo, per cui il flusso tipico di rilascio potrebbe essere semplicemente

```
make publish
```

Il nuovo contenuto dovrebbe apparire negli url opportuni, che sono quelli pre-compilato nel file ‘README.md‘ del repositorio.

Cronologia delle versioni A.

- Draft iniziale.

Bibliography