Accessible UC Berkeley Thesis Template

by

Oski Bear

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy
in
Structural Engineering
in the
Graduate Division
of the
University of California, Berkeley

Committee in charge:

Professor Ludwig von Sather, Co-chair
Professor Golden C. Bear, Co-chair
Professor Josephine Doe-Bancroft
Professor Sproul B. Strawberry

Spring 2026

Accessible UC Berkeley Thesis Template

Abstract

Accessible UC Berkeley Thesis Template

by

Oski Bear

Doctor of Philosophy in Structural Engineering

University of California, Berkeley

Professor Ludwig von Sather, Co-chair

Professor Golden C. Bear, Co-chair

Template for producing UC Berkeley theses and dissertations as tagged PDF files that meet WCAG 2.1 AA accessibility requirements. Compiles with the `wcagthesis` class, a `report`-based replacement for the `ucbthesis` class that preserves the Graduate Division's required formatting and layout while supporting LaTeX PDF tagging.

To whom this work is dedicated.

Contents

Acknowledgments

This template is adapted, with gratitude, from the UC Berkeley thesis template maintained by Paul Vojta (Vojta, 2019).

Chapter 1

About This Template

1.1 Purpose and Origin

This is a template for writing a University of California, Berkeley thesis or dissertation in LaTeX and compiling it to a *tagged* PDF based on the WCAG 2.1 AA accessibility standard.

As of April 27, 2027, a new UCOP (UC Office of the President) accessibility policy will come into effect stating that digital content produced by the University must conform to the Web Content Accessibility Guidelines, version 2.1, at conformance level AA. Guidance is documented online:

- UCOP guidance on the UC IT Accessibility Policy (ITAP) at `https://digitalaccessibility.ucop.edu/` (The Regents of the University of California, 2026a)
- UCOP Electronic Accessibility policy overview page at `https://www.ucop.edu/electronic-accessibility/initiative/policy.html` (The Regents of the University of California, 2026b)
- UC Berkeley Digital Accessibility Program summary and guide at `https://dap.berkeley.edu/websites/what-digital-accessibility` (University of California, Berkeley, 2026)
- UC Berkeley guidance on Accessibility in Teaching and Learning at `https://rtl.berkeley.edu/resources/accessibility-teaching-learning`
- World Wide Web Consortium guidelines at `https://www.w3.org/WAI/standards-guidelines/wcag/` (World Wide Web Consortium, 2026)

- Overleaf blog post regarding accessible PDFs at `https://www.overleaf.com/blog/accessible-pdfs-with-latex` (Overleaf, 2026a)
- Overleaf guide to creating accessible PDFs in LaTeX at `https://docs.overleaf.com/writing-and-editing/creating-accessible-pdfs` (Overleaf, 2026b)
- Adobe's guide to creating and verifying PDF accessibility in Acrobat Pro at `https://helpx.adobe.com/acrobat/using/create-verify-pdf-accessibility.html` (Adobe, 2026)

A major conformance criterion for a PDF is that the file is *tagged*: it carries a logical structure tree (headings, paragraphs, lists, tables, figures with alternative text, reading order) that assistive technology can navigate. Tagging capabilities have been developed as part of a major effort called the LaTeX Tagged PDF Project (The LaTeX Project, 2026).

Since its creation circa 2014, Paul Vojta's UC Berkeley Thesis Template has been used widely for doctoral dissertations written in LaTeX. Vojta's template is built on the `ucbthesis` class, which uses the `memoir` document class (Vojta, 2019). The `memoir` class does not support LaTeX's PDF-tagging code. This template therefore uses a new class, `wcagthesis`, that is built on the standard `report` class[1]. It reproduces the Graduate Division's required layout and formatting from `ucbthesis`, while remaining compatible with tagging.

Note on LaTeX's evolving PDF-tagging code. LaTeX's PDF-tagging code is under active development, and its behavior changes from release to release. There are several known limitations of the current code that are being fixed by the LaTeX Tagged PDF Project (see `https://github.com/latex3/tagging-project/issues`).

Relatedly, in this template there are a few manual patches in `Style/Tagging.tex` that make certain components such as figures and tables pass Adobe Acrobat's accessibility checker. Because LaTeX's PDF-tagging code changes so rapidly, these patches are version-sensitive: a future LaTeX release may make them unnecessary or require them to be updated.

This template was developed for the WCAG 2.1 AA guidelines, current as of August 21, 2026, and for TeX Live 2026 / LaTeX 2026-06-01. Reasonable effort was made to comply with these guidelines, and the template was verified to pass the accessibility checks in Adobe Acrobat Pro. However, the authors cannot guarantee that the template meets future guidelines, that document components not represented in the template will be tagged correctly for accessibility, or that its tagging commands remain compatible with future versions of LaTeX.

[1] To toggle built-in stylistic options, authors are referred to the LaTeX Project's documentation on the `report` document class `https://mirror.clarkson.edu/ctan/macros/latex/base/classes.pdf`.

1.2 Template Files

1.2.1 `main.tex`

The file `main.tex` is the main file that is passed to the LaTeX compiler. It includes all the other `.tex` files that comprise the final document.

1.2.2 `setup.tex`

The file `setup.tex` exposes macros that set up the document.

Disable Tagging. Tagging can be disabled by changing the command `\Taggingtrue` to `\Taggingfalse` at the top of `setup.tex`. This is useful for drafting, since compiling will take significantly longer with `\Taggingtrue`. However, **the final submitted document must be compiled with `\Taggingtrue`**.

1.2.3 `wcagthesis.cls`

The file `wcagthesis.cls` implements the `wcagthesis` document class. Authors will not generally need to modify this file.

Chapter 2

Producing a Tagged, Accessible PDF

2.1 Configuring LaTeX's PDF-tagging code

All PDF tagging in modern LaTeX is switched on by a single command, `\DocumentMetadata`, which *must* appear before `\documentclass`. In this template it lives at the top of `setup.tex`, inside the `\ifTagging` branch:

```
\DocumentMetadata{
  lang         = en-US,
  pdfstandard  = ua-2,
  pdfstandard  = a-4,
  tagging-setup = {
    modules = {phase-III, title, table, firstaid, names, toc, sec},
    table/tagging       = presentation,
    table/header-rows   = {1},
  }
}
```

Setting `\DocumentMetadata` with a `tagging-setup` key loads and activates LaTeX's PDF-tagging code (`tagpdf`) automatically; you do not load any package yourself. The `lang` key sets the document language, and the `pdfstandard` keys declare conformance with PDF/UA-2 and PDF/A-4. Compile the document with **LuaLaTeX** on TeX Live 2025 or newer.

Once tagging is active, the PDF-tagging code handles most structure for you: headings and paragraphs, the table of contents, cross-references and citations, lists, tables (with the header row declared above), and mathematics.

2.2 What you must still do by hand

Four things require author action:

1. **Document metadata.** Set the title, author, and language in the `\hypersetup` block of `main.tex` (Section 1.2.1). A tagged PDF must declare a title and a language.
2. **Alternative text on every figure.** Give each `\includegraphics` (and each `tikzpicture`) an `alt=` option describing the image for a screen reader; see Section 2.3.
3. **Artifacts for decorative content.** Mark purely decorative or duplicative graphics so assistive technology skips them; see Section 2.3.
4. **Remediation in Acrobat.** Compile, then verify and repair the tag structure in Adobe Acrobat; see Section 2.4.

Beyond these, consult the LaTeX Tagging Project website (The LaTeX Project, 2026), in particular its documentation and the *Using LaTeX to Produce Accessible PDF* guide, for current best practices. Those pages are updated often and should be treated as the authoritative, up-to-date guidance, which may supersede the recommendations in this template.

2.3 Alternative text and artifacts

Text alternatives must be provided for any non-text content.

Alt text. Add the `alt=` key directly in the options of `\includegraphics`:

```
\includegraphics[width=0.85\linewidth,
  alt={Deformed shape of the enhanced beam model, shown as a warped
       tubular mesh.}]{Figures/veux-frame.png}
```

For an image that is really rendered text (a scanned equation, say), use `actualtext=` instead, so the text is exposed verbatim.

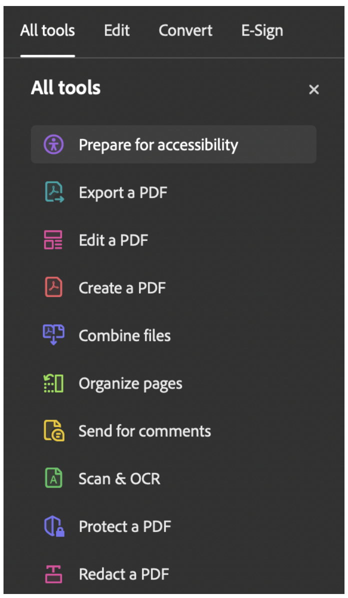
Artifacts. A graphic that carries no information (e.g. a background flourish or a box whose content is already stated in the running text) should be an *artifact* so it is not announced by a screen reader. This template provides an `Artifact` environment that suspends tagging around its contents:

```
\begin{Artifact}
  ... decorative material ...
\end{Artifact}
```

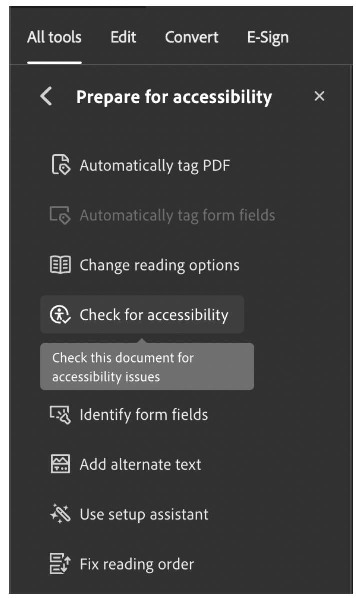
2.4 Checking and fixing accessibility in Adobe Acrobat

LaTeX produces a relatively well-tagged file, but you should **verify and repair the compiled result early and often in Adobe Acrobat Pro**. This will help you resolve WCAG compliance issues well before the filing deadline.

Open the compiled PDF in Acrobat and open the *Prepare for accessibility* tool from the *All tools* list (Figure 2.1a). From that tool, choose *Check for accessibility* (Figure 2.1b).



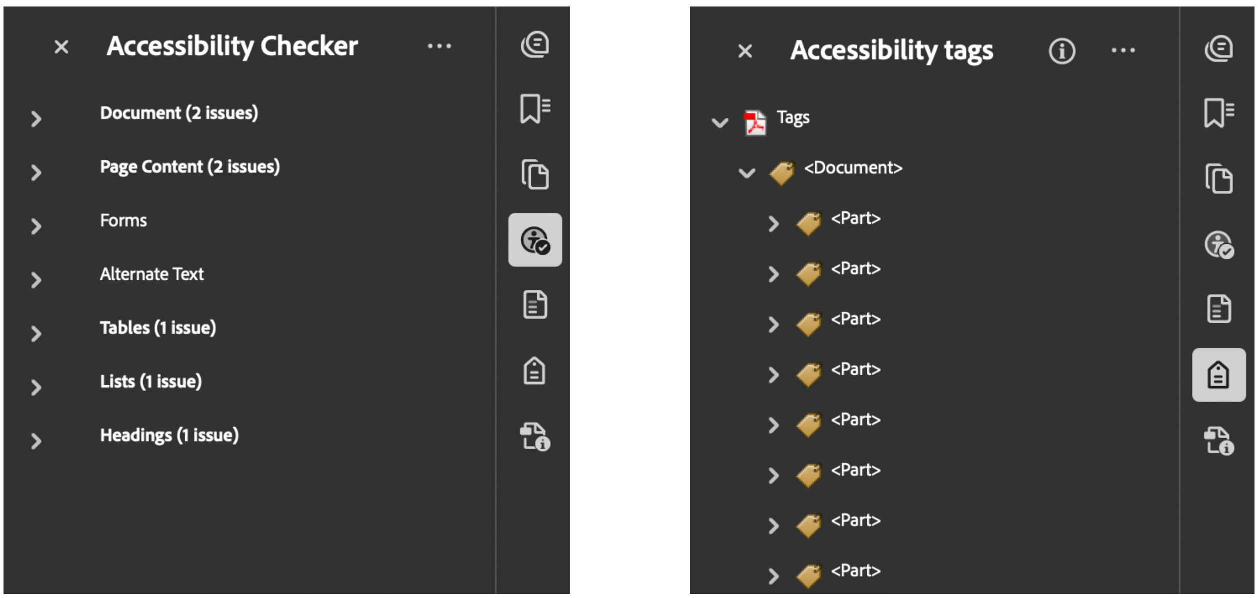
(a) The All tools list.



(b) The Prepare for accessibility panel.

Figure 2.1: The Adobe Acrobat *Prepare for accessibility* tool and its *Check for accessibility* action.

The checker reports each rule as passed, failed, or needing manual review in the *Accessibility Checker* panel (Figure 2.2a); right-click a failed item to fix it or to jump to the offending content. Right-clicking an item and choosing *Explain* opens the relevant page of Adobe's accessibility guide (Adobe, 2026). Use the *Tags* panel (Figure 2.2b) to confirm the reading order, retag an element, add or edit alt text on a `<Figure>` tag, or mark stray content as an artifact. Typical findings for a LaTeX PDF are a missing document title, a figure without alt text, or a table missing a header row.



(a) The Accessibility Checker results panel.

(b) The Tags panel (logical structure tree).

Figure 2.2: Reviewing results and structure: the Accessibility Checker and the Tags panel.

Chapter 3

Example Document Elements

Common elements are collected here so that their tagged output can be examined in Acrobat. The content is illustrative only and is adapted from Perez (2026).

3.1 Figures

Every figure declares alt text on its `\includegraphics` call. Figure 3.1 shows an example.

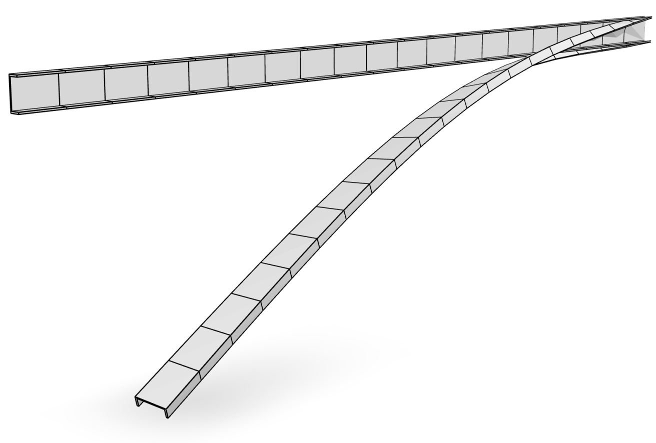


Figure 3.1: Deformed shape of an enhanced beam model.

3.2 Tables

The header row of Table 3.1 is declared to LaTeX's PDF-tagging code through `table/header-rows={1}` in `setup.tex`, so it is tagged as a real table header (`<TH>`) rather than hidden as an artifact.

Table 3.1: Parameters identifying a trace in the Saint Venant class.

Block	Size	Units	Description
$\widehat{G\mathbf{A}}_\gamma$	$\mathbb{R}^{2\times 2}$	L^2	Trace-consistent shear rigidity
$\boldsymbol{\Xi}_\kappa$	$\mathbb{R}^{2\times 2}$	L^1	Curvature–flexure coupling
$\boldsymbol{\rho}_\gamma$	$\mathbb{R}^2$	L	Shear–torsion coupling
$\boldsymbol{\rho}_\varkappa$	$\mathbb{R}^2$	L	Twist–flexure coupling
$\boldsymbol{\varsigma}_\kappa$	$\mathbb{R}^2$	$\sim$	Curvature–torsion coupling
$\boldsymbol{\varsigma}_\epsilon$	$\mathbb{R}^2$	$\sim$	Axial–flexure coupling

3.3 Equations

The following is an example of aligned equations with sequential equation numbers (refer to Appendix A for an example of a standalone equation):

$$\vartheta(x) = \frac{T_0}{\widehat{GJ}_\vartheta} x + \frac{c_1}{\lambda} \sinh(\lambda x) + \frac{c_2}{\lambda} \cosh(\lambda x) + c_3. \tag{3.3.1}$$

$$W(x) = \widetilde{EJ}_\mathrm{w} \left(1 + \frac{\widehat{GJ}_\vartheta \, J_\eta}{\widetilde{EJ}_\mathrm{w}^2}\right) \lambda \left[c_1 \sinh(\lambda x) + c_2 \cosh(\lambda x)\right], \tag{3.3.2}$$

$$V(x) = -\widehat{GJ}_\vartheta \left[c_1 \cosh(\lambda x) + c_2 \sinh(\lambda x)\right]. \tag{3.3.3}$$

3.4 Propositions, Theorems, Remarks

Theorem-like environments are tagged as ordinary structured text.

Proposition 3.4.1 (Equilibrium compatibility). *Let $\mathcal{T}$ be a Class I trace, let $\mathbf{C}_T := \mathbf{K}_s$, and assume that $\bar{\mathbf{C}}_T$ is prismatic. Then the section equilibrium relation holds for every*

Saint Venant solution if and only if

$$[\bar{\mathbf{G}}, \mathbb{J}] = \mathbf{0}, \qquad \bar{\mathbf{G}} := \bar{\mathbf{C}}_T \mathbf{C}_T^{-1}. \tag{3.4.1}$$

Proof. ... □

Remark 3.4.1. The hockling problem arises from practical considerations for the design of propeller shafts in large ships (O'Reilly, 2017).

3.5 Algorithms

Algorithm 1 Fiber discretization $\mathcal{F}(b, d)$.

function $\mathcal{F}(b, d, n_x, n_y)$ ▷ *Stage 1: quadrilateral mesh over* $[-b/2, b/2] \times [-d/2, d/2]$
  build node coordinates $\mathbf{x}$ and connectivity $\mathbf{C}$ ▷ ...
  ▷ *Stage 2: Gauss integration*
  $\mathcal{Q} \leftarrow$ tensor-product Gauss points and weights ▷ ...
  $\mathcal{S} \leftarrow \emptyset$
  for all $e \in \mathbf{C}$ **do**
    map each quadrature point to a fiber (y_k, z_k) of area A_k ▷ ...
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{(y_k, z_k, A_k)\}$
  end for
  return $\mathcal{S} = \{(y_k, z_k, A_k)\}_{k=1}^{N}$
end function

3.6 Boxes

Box 3.1 summarizes the governing equations of a Cosserat rod with warping (Simo & Vu-Quoc, 1991). The box is wrapped in the `Artifact` environment, since it merely restates equations that also appear in the tagged text.

Box 3.1: Geometrically Exact Frame Model

Equilibrium

$$\begin{cases} (\boldsymbol{\Lambda n})' + \boldsymbol{f}_n = \mathbf{0} \\ (\boldsymbol{\Lambda m})' + \boldsymbol{x}' \times (\boldsymbol{\Lambda n}) + \boldsymbol{f}_m = \mathbf{0} \\ \boldsymbol{w}' - \boldsymbol{v} + \boldsymbol{f}_{\mathrm{w}} = \mathbf{0} \end{cases}$$

Kinematics

$$\boldsymbol{\gamma} = \boldsymbol{\Lambda}^\top \boldsymbol{x}' - \mathbf{i} \qquad \text{and} \qquad \boldsymbol{\kappa} = \left(\boldsymbol{\Lambda}^\top \boldsymbol{\Lambda}'\right)^\vee$$

Chapter 4

References

Adobe. (2026). *Create and verify PDF accessibility (Acrobat Pro)*. Adobe. Retrieved August 12, 2026, from https://helpx.adobe.com/acrobat/using/create-verify-pdf-accessibility.html

Chern, C., & O'Reilly, O. M. (2026). On complex modal inverse analysis: The cautionary tale of the single surviving mode. *Journal of Applied Mechanics*, 1–5. https://doi.org/10.1115/1.4072378

McCallen, D., Pitarka, A., Tang, H., Nakata, R., Mosalam, K. M., Petrone, F., Günay, S., & Perez, C. (2025). An open-access simulated earthquake ground-motion database for an M7 Hayward Fault earthquake in the San Francisco Bay Region. *Earthquake Spectra*, *41*(3), 2560–2597. https://doi.org/10.1177/87552930251340960

O'Reilly, O. M. (2017). *Modeling nonlinear problems in the mechanics of strings and rods*. Springer. https://doi.org/10.1007/978-3-319-50598-5

Overleaf. (2026a, January). *Accessible PDFs with LaTeX*. Overleaf. Retrieved August 24, 2026, from https://www.overleaf.com/blog/accessible-pdfs-with-latex

Overleaf. (2026b, August). *Creating accessible PDFs in LaTeX*. Overleaf. Retrieved August 24, 2026, from https://docs.overleaf.com/writing-and-editing/creating-accessible-pdfs

Perez, C. M. (2026, June). *Nonlinear modeling of frame members for rapid infrastructure assessment* [Ph.D. dissertation]. University of California, Berkeley.

Simo, J., & Vu-Quoc, L. (1991). A Geometrically-exact rod model incorporating shear and torsion-warping deformation. *International Journal of Solids and Structures*, *27*(3), 371–393. https://doi.org/10.1016/0020-7683(91)90089-X

The LaTeX Project. (2026, June). *The LaTeX tagged PDF project*. The LaTeX Project. Retrieved August 12, 2026, from https://latex3.github.io/tagging-project/

The Regents of the University of California. (2026a). *Digital accessibility resources*. University of California Office of the President. Retrieved August 24, 2026, from `https://digitalaccessibility.ucop.edu/`

The Regents of the University of California. (2026b). *Electronic accessibility policy / ucop*. University of California Office of the President. Retrieved August 24, 2026, from `https://www.ucop.edu/electronic-accessibility/initiative/policy.html`

University of California, Berkeley. (2026). *What is digital accessibility?* Digital Accessibility Program. Retrieved August 12, 2026, from `https://dap.berkeley.edu/websites/what-digital-accessibility`

Vojta, P. (2019, July). *UC Berkeley thesis template*. Overleaf. Retrieved August 12, 2026, from `https://www.overleaf.com/latex/templates/uc-berkeley-thesis-template/mfzmtxfqvtxx`

World Wide Web Consortium. (2026, May). *WCAG 2 overview*. W3C Web Accessibility Initiative. Retrieved August 12, 2026, from `https://www.w3.org/WAI/standards-guidelines/wcag/`

Appendix A

Supplementary Material

Equation (A.0.1) is an example of a standalone equation that serves as a complement to the aligned equations in the main text (Equations (3.3.1) to (3.3.3)). It describes the retrieval of a real-valued mode shape from an identified complex-valued mode $\boldsymbol{\varphi}_1$ and its corresponding eigenvalue $\lambda_1 = a_1 + ib_1$ (Chern & O'Reilly, 2026).

$$e^{i\phi_1}\boldsymbol{\varphi}_1 = \begin{bmatrix} 0 \\ \frac{\mathrm{sgn}(a_1)}{\Delta}\sqrt{a_1^2 + b_1^2} \end{bmatrix}, \text{ where } \phi_1 = \arctan\left(\frac{b_1}{a_1}\right). \tag{A.0.1}$$