

Accessibility Recommendations for LaTeX Documents

eSAIL @ TAMU - Instructional Design + Accessibility Team

October 2026

1 Disclaimer and Context

These accessibility recommendations are based on the knowledge gathered by the eSAIL team at Texas A&M and the work of the LaTeX Tagging Project [1], [2]. They represent current best practices for creating PDF/UA-2 documents with LuaLaTeX. Tagging support in LaTeX is under active development, so check the project's documentation for changes.

Requirements: This document requires the LaTeX release of 2025-11-01 or newer and must be compiled with LuaLaTeX. On Overleaf, select *Rolling TeX Live* in the project settings; the default TeX Live version is too old for these settings.

Instructional Note on Metadata: In the source code of this document, the `\DocumentMetadata` command appears before `\documentclass`. Its `tagging=on` key replaces the older `testphase` setups and loads all the code needed to tag paragraphs, lists, tables, and math [1]. The `check-tagging-status` key writes a report to the log file listing whether each package you load is compatible with tagging.

1.1 Build Information

The list below is generated when the document compiles, using the `\BuildInfo` command defined in the preamble. Use it to confirm the requirements above: the engine should be LuaLaTeX, the LaTeX release 2025-11-01 or newer, tagging on, and the PDF version 2.0.

- **Engine:** This is LuaHBTeX, Version 1.24.0 (TeX Live 2026)
- **LaTeX release:** 2026-06-01
- **Tagging:** on
- **PDF version:** 2.0
- **unicode-math:** 2023/08/13 v0.8r Unicode maths in XeLaTeX and LuaLaTeX
- **Main font:** "name:STIX Two Text:mode=node;script=latn;language=dflt;+tlig;" at 10.95pt
- **Compiled on:** October 23, 2026

2 Basic Document Structure and Headings

2.1 Template Style

This example uses IEEE-style conventions for captions and the Times-like STIX Two fonts for text and math. Under LuaLaTeX these are OpenType fonts that embed as Unicode, so text and math can be extracted and read aloud correctly.

2.2 Sample Paragraph and Tagging

Because the `\section` and `\subsection` commands are used, the PDF tagging system maps these lines to semantic heading tags. Screen readers use these headings to navigate the document.

2.3 Lists (Semantic Bullets and Steps)

We use the standard `itemize` and `enumerate` environments. The `tagging=on` setting automatically converts these into L (List), LI (List Item), Lbl (Label), and LBody (List Body) tags, which screen readers need to announce lists correctly.

Unordered list:

- Use `\begin{itemize}` for bullet lists.
- Screen readers announce lists and list items.

Ordered list:

1. First step in the process.
2. Second step in the process.

2.4 Links with Meaningful Link Text

For example, visit the Texas A&M University website instead of writing out the raw URL.

3 Data Presentation (Accessible Tables and Figures)

3.1 Why We Use Global Header Settings

In the preamble, we set `table/header-rows=1`. This tells the compiler to tag the first row of *every* table as a header row (TH). Without header cells, screen readers cannot associate the data in row 5 with the label in row 1.

Because this is a global default, override it for tables that are different: use `\tagpdfsetup{table/header-rows={1,2}}` before a table with two header rows, or `\tagpdfsetup{table/tagging=presentation}` before a table used only for layout.

TABLE I: Selected Material Properties for Common Engineering Elements

Element	Density (g/cm ³)	Melting point (°C)	Thermal conductivity (W/m·K)	Young's modulus (GPa)
Aluminum (Al)	2.70	660	237	70
Steel (Fe)	7.87	1538	80.4	210
Copper (Cu)	8.96	1085	401	110
Titanium (Ti)	4.51	1668	21.9	116
Tungsten (W)	19.25	3422	173	411

3.2 Example: Data Presentation Figure

This figure demonstrates the “long description” pattern: short alt text that states the figure’s purpose, plus a caption link to a full description in the appendix.

Alt text should describe what the reader is meant to learn from the figure, not only what it looks like.

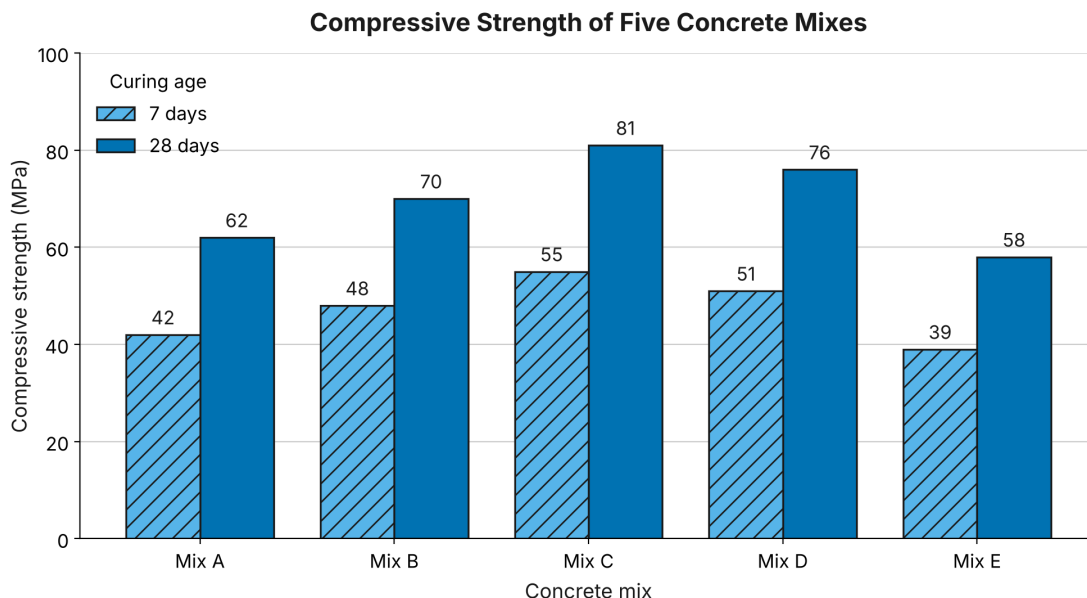


Fig. 1: Compressive Strength of Five Concrete Mixes, illustrative data (**Extended Description**)

4 Mathematics and Equations

4.1 The End-to-End MathML Workflow

Compiling with LuaLaTeX and loading `unicode-math` lets LaTeX generate MathML for every equation automatically. In the preamble, we enabled `math/setup={mathml-SE, mathml-AF}`:

- **mathml-SE (Structure Elements)**: Places the MathML directly in the PDF tag tree, a PDF 2.0 feature.
- **mathml-AF (Associated Files)**: Attaches the MathML to each formula as an embedded file.

PDF readers and assistive technologies differ in which method they support, so we include both. Users with NVDA and the **MathCAT** add-on can then read and navigate the equations below term by term [4].

$$E = mc^2 \tag{1}$$

4.2 Multi-line Example Using Align

$$T(s) = \frac{Y(s)}{R(s)} \quad (2)$$

$$= \frac{G(s)}{1 + G(s)G_c(s)} \quad (3)$$

5 Additional Text Features

5.1 Code Blocks (Verbatim)

We enabled the experimental `verbatim-mo` module in the tagging setup. It tags the content below as code, so screen readers announce punctuation such as parentheses, brackets, and slashes that they would otherwise skip in plain text.

```
def calculate_grade(score, max_score=100):  
    if max_score <= 0:  
        return 0.0  
    return (score / max_score) * 100
```

6 Complex Visual Content

For complex images, we provide a brief `\caption` for sighted users and an explicit, linked long description in the appendix.

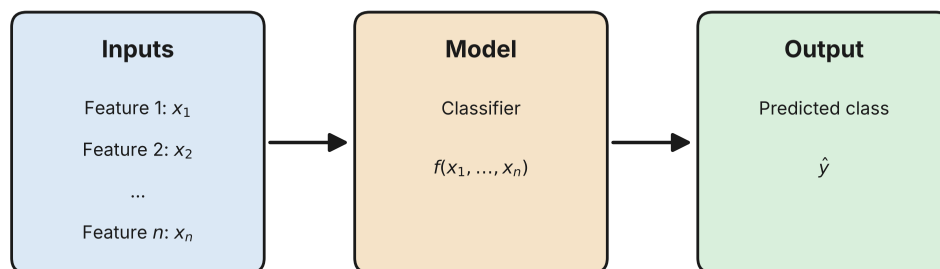


Fig. 2: Example Classifier Diagram (**Extended Description**)

For a full, narrative-style description of the classifier diagram, see Appendix A.2.

6.1 Decorative Images (Artifacts)

Not every image carries information. Decorative elements such as borders, background textures, and section dividers should be marked as *artifacts* with the `artifact` key instead of being given alt text. Screen readers then skip them, so users are not interrupted by descriptions like “decorative line.”

The divider below is an artifact:

`\includegraphics[width=0.6\linewidth, artifact]{images/section-divider.png}`

Use an artifact only when removing the image would lose no information. If an image conveys meaning, give it alt text instead.



7 Layout Practices: What to Avoid

Avoid layout hacks like repeated `\\` or `\vspace`. Let LaTeX handle spacing so the reading order is preserved.

8 Final Check and Validation

8.1 PDF/UA-2 vs. Reality

This document targets **PDF/UA-2** [3], [5]. Be aware of the limits of current tools:

- **Institutional checkers:** Ally, Panorama, and the Adobe Acrobat accessibility checker do not support PDF 2.0 or PDF/UA-2 (vendor-confirmed). They may flag valid MathML or modern table structures as errors, for example reporting equations as missing alt text because they expect the older PDF/UA-1 approach. They may even score this document lower than a less accessible PDF/UA-1 file, so a low score from these tools does not mean the document is inaccessible.
- **PAC:** Fully supports PDF 1.7 documents only; PDF/UA-2 support is limited, with full support planned for a future release.
- **veraPDF:** Validates PDF/UA-2, but only the machine-verifiable rules [6]. Requirements that need human judgment, such as whether alt text is meaningful or reading order is logical, are outside any automated validator.
- **The real test:** Read the document with **NVDA + MathCAT**. If the headings, lists, tables, and equations are read and navigable, the content is accessible regardless of an automated score.

8.2 Recommended Validation Steps

1. **Compile log:** Check the `check-tagging-status` report for incompatible packages, and look for warnings about unknown keys.
2. **veraPDF:** Validate against the PDF/UA-2 profile.
3. **Screen reader review:** Test with NVDA and MathCAT so headings, lists, tables, and equations are read correctly.
4. **Reading order review:** Verify the logical order in Acrobat's Tags panel.

9 References

1. U. Fischer and F. Mittelbach, “News from the LaTeX Tagged PDF project: 2025,” *TUGboat*, vol. 46, no. 2, 2025.
2. LaTeX Project Team, “Using LaTeX to produce accessible PDF,” *LaTeX Tagging Project*. [Online]. Available: <https://tagging-project.latex-project.org/documentation/usage-instructions>
3. ISO, “Document management applications — Electronic document file format enhancement for accessibility — Part 2: Use of ISO 32000-2 (PDF/UA-2),” ISO 14289-2:2024, Geneva, Switzerland, 2024.
4. N. Soiffer, “MathCAT: Math Capable Assistive Technology,” *GitHub Repository*. [Online]. Available: <https://github.com/NSoiffer/MathCAT>
5. PDF Association, “Well-Tagged PDF (WTPDF),” *PDF Association Technical Specification*, 2024.
6. veraPDF Consortium, “veraPDF: Industry Supported PDF/A and PDF/UA Validation.” [Online]. Available: <https://verapdf.org>
7. LaTeX Project Team, “Tagging Status of LaTeX Packages and Classes.” [Online]. Available: <https://latex3.github.io/tagging-project/tagging-status/>

10 Credits

Developed and maintained by Gerry Pedraza and Christine Roach, Engineering Studio for Advanced Instruction and Learning (eSAIL), Texas A&M University.

Research and testing contributions: Osvaldo Misael Cortes, Lydia K. Harrison, and Vinayaka Madhukeshwar Hegde.

This work builds on the LaTeX Tagging Project by the LaTeX Project Team. Template repository: <https://github.com/gerrypedraza/accessibility>. Suggestions and problem reports are welcome as issues on the repository.

A Appendix: Long Descriptions

Extended descriptions of the figures in this document.

A.1 Extended Description: Compressive Strength Chart

Purpose. Fig. 1 compares how strong five concrete mixes become as they cure. It shows which mix is strongest and how much strength each mix gains between an early test (7 days) and the standard design test (28 days). The values are illustrative, not measured data.

Layout. The figure is a grouped bar chart titled “Compressive Strength of Five Concrete Mixes.” The horizontal axis, labeled “Concrete mix,” lists five mixes from left to right: Mix A, Mix B, Mix C, Mix D, and Mix E. The vertical axis, labeled “Compressive strength (MPa),” runs from 0 to 100 megapascals, with tick marks and light gray horizontal gridlines every 20 MPa. Each mix has a pair of side-by-side bars. The left bar of each pair shows strength after 7 days of curing and is light blue with diagonal hatching. The right bar shows strength after 28 days and is solid dark blue. The pattern, not only the color, distinguishes the two series. A legend titled “Curing age” in the upper left identifies the two bar styles, and every bar has its exact value printed above it.

Data. The table lists every value in the chart, plus the gain from 7 to 28 days.

Mix	7 days (MPa)	28 days (MPa)	Gain (MPa)
Mix A	42	62	20
Mix B	48	70	22
Mix C	55	81	26
Mix D	51	76	25
Mix E	39	58	19

Trends and comparisons.

- Every mix is stronger at 28 days than at 7 days. The gains range from 19 MPa (Mix E) to 26 MPa (Mix C).
- The ranking is the same at both ages. From strongest to weakest: Mix C, Mix D, Mix B, Mix A, Mix E.
- Mix C is the strongest mix at both ages, with 55 MPa at 7 days and 81 MPa at 28 days. Mix D is close behind, with 51 and 76 MPa.
- Mix E is the weakest at both ages, with 39 and 58 MPa. At 28 days, Mix C is 23 MPa (about 40%) stronger than Mix E.
- For every mix, the 7-day strength is about two-thirds (67 to 69%) of the 28-day strength, so the 7-day test is a consistent early predictor of final strength.
- Averaged across all five mixes, strength rises from 47 MPa at 7 days to about 69 MPa at 28 days.

[\[Return to Figure 1 in main content\]](#)

A.2 Extended Description: Classifier Diagram

Purpose. Fig. 2 shows the basic structure of a classifier: information flows from a set of input measurements, through a model, to a single predicted category. It is meant to convey the idea that a classifier is a function, not the details of any particular algorithm.

Layout. The diagram is read from left to right. It contains three rounded rectangular boxes of equal size, side by side, each with a bold title at the top. Two thick black arrows connect them: one from the

first box to the second, and one from the second box to the third. The boxes have different light fill colors (blue, orange, and green), but the colors carry no meaning; the titles and arrows convey the structure.

The three boxes, in reading order.

1. **Inputs** (left, light blue): a vertical list of the features describing one item to be classified. It reads “Feature 1: x_1 ,” “Feature 2: x_2 ,” then an ellipsis indicating that more features follow, and finally “Feature n : x_n .” There can be any number of features, n , such as the measurements of a sample or the attributes of a record.
2. **Model** (center, light orange): the word “Classifier,” with the function $f(x_1, \dots, x_n)$ written below it. The notation shows that the model takes all n feature values together as its arguments. The arrow from Inputs into this box shows the features being passed to the model.
3. **Output** (right, light green): the words “Predicted class,” with the symbol $\hat{y}$ (read “y hat”) below them. The hat marks it as an estimate made by the model rather than a known true value. The arrow from Model into this box shows the model producing its prediction.

Main idea. Reading the three boxes in order gives the relationship $\hat{y} = f(x_1, \dots, x_n)$: a classifier maps many input features to one predicted category. For example, the features might be the measurements of a flower, and the predicted class its species.

[Return to Figure 2 in main content]

[End of Document]