AI Templates

Contents

[1 General Principles for AI-Friendly Templates 2](#_Toc207526379)

[1.1 Consistency > Complexity 2](#_Toc207526380)

[1.1.1 Headings Consistency Example 2](#_Toc207526381)

[1.2 Tables Consistency Example 2](#_Toc207526382)

[2 **3. Ratings/Assessments Example** 4](#_Toc207526383)

[2.1 Use Explicit Labels 5](#_Toc207526384)

[2.2 Machine-Readable Tables 5](#_Toc207526385)

[2.3 Inline Metadata 5](#_Toc207526386)

[2.4 Separation of Narrative vs Data 5](#_Toc207526387)

[2.5 Versioning 5](#_Toc207526388)

[Table 1 Incorrect table format - missing values 2](#_Toc207526367)

[Table 2 Incorrect table format - merged cells 3](#_Toc207526368)

[Table 3 Correctly Presented Table Format 3](#_Toc207526369)

# General Principles for AI-Friendly Templates

## Consistency Over Complexity

Keep headings, tables, and sections consistent across documents. AI relies on patterns; variation increases error rates.

**AI models perform best when they can rely on predictable patterns** in the structure of your documents. If headings, tables, and labels are always in the same place and use the same wording, the model doesn’t have to “guess” what you mean.

Complexity (like merged cells, inconsistent headers, or changing section names) creates ambiguity and increases the chance of misinterpretation.

Why This Matters for AI:

* **Pattern matching**: Models like GPT can use section anchors like “## Financials” to locate the right part every time.
* **Reduced hallucination**: When the model doesn’t need to infer whether “Financial Snapshot” = “Financials”, the risk of wrong outputs drops.
* **Easier automation**: If later you want to connect outputs to a database (e.g., populate Power BI or an internal system), consistent structures make mapping trivial.

Rule of Thumb:

* Same headings, same order, same wording.
* Flat tables over merged/complex tables.
* Labels should act like database field names.

### Headings Consistency Example

Inconsistent headings across documents creates confusion. For example, the financial section could be worded in many ways, like:

* Report A is worded as: “Company Financial Overview”
* Report B is worded as: “Financial Performance Snapshot”
* Report C is worded as: “Key Financial Data”

AI has to “learn” that these all mean *the same thing*. It doesn’t always get this right, leading to errors. To avoid these errors, consistent headings across all documents should be used, for example:

* Always use: “1 Financials”
* Subsections: “1.1 Income Statement”, “1.2 Balance Sheet”, “1.3 Cash Flow”

Now AI can **anchor on the same labels every time** and map outputs correctly. Note, the above applies the ISO 2145:1978 – Numbered Sections in Written Documents standard.

### Tables Consistency Example

Complex Table (merged cells, inconsistent years) create issues in the use of AI, for example:

Table 1 Incorrect table format - missing values

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **2024** | **2023** | **2022** |
| Revenue | 500 | 450 | 420 |
| EBITDA/NPAT | 200 | 180 |  |
| Margin % | 40% |  | 39% |

Problems:

* Combined row (“EBITDA/NPAT”) confuses AI.
* Missing cells break pattern.
* Alignment isn’t consistent.

Table 2 Incorrect table format - merged cells

|  |  |  |
| --- | --- | --- |
| **Area** | **Item** | **FY24 (AUDm)** |
| Income Statement | Revenue | 100 |
| EBITDA | 45 |
| NPAT | 35 |
| Balance Sheet | Total Assets | 1000 |
| Total Liabilities | 800 |
| Current Ratio | 2.9 |

Problems:

* Merged cells break the row-by-row pattern
* AI can mis-align the rows, e.g. EBITDA might appear without being allocated to the Income Statement

Table 3 Flat Table Format

|  |  |  |
| --- | --- | --- |
| **Area** | **Item** | **FY24 (AUDm)** |
| Income Statement | Revenue | 100 |
| Income Statement | EBITDA | 45 |
| Income Statement | NPAT | 35 |
| Balance Sheet | Total Assets | 1000 |
| Balance Sheet | Total Liabilities | 800 |
| Balance Sheet | Current Ratio | 2.9 |

Here:

* Each metric has its own column.
* No merged cells.
* Same number of values for each year.

### Ratings/Assessments Example

Inconsistent Field Labels

* Report A: *“Internal Risk Grade: BBB+”*
* Report B: *“Macquarie Rating: BBB+ (internal)”*
* Report C: *“Risk Assessment → BBB+”*

Standardized Field Label

* Always: *“Internal Rating: BBB+”*

This ensures AI extraction yields one reliable field.

## Use Explicit Labels

Every section and table should have a clear, descriptive title and avoid generic numbering-only references such as “Table 1” or “Section A”. Instead, ensure that the heading or caption tells both the human reader and the AI what the content relates to.

Why This Matters

* For Humans: Readers can immediately understand what the section or table covers without needing to cross-reference another list or appendix.
* For AI: Models don’t have to infer context. A heading like “Income Statement (AUD millions, FY24)” gives enough signal that this is financial statement data, denominated in AUD, for fiscal year 2024.
* For Automation: Downstream tools (e.g., Python parsers, Power BI, or GPT prompts) can grab a table by name (e.g., “Balance Sheet”) instead of just “Table 2”, which is ambiguous if documents differ.

Key Principle

* Always assume that the table or section might be copied, extracted, or read out of context. If the label is explicit, the meaning survives outside the original document.

### Good vs Bad Heading Examples

A heading like “Section 3.1” is not descriptive and doesn’t indicate what is included in the section. A better heading would be “Section 3.1 - Prior 12 Months Activities”. An example of what a good table of contents would look like is:

1 Introduction

2 Methodology

2.1 Counting techniques

2.1.1 Manual procedures

2.1.1.1 Counting apples

2.1.1.2 Counting oranges

2.1.2 Automatic methods

2.2 Quality control

3 Results

4 Related work

4.1 Bean counting

4.2 Sheep counting

5 Conclusions

Note, this aligns with the intent of ISO 2145:1978 - Numbered Sections in Written Documents, which recommends hierarchical numbering for structure, but you should also apply meaningful labels alongside numbering.

### Good vs Bad Caption Examples

When labelling tables and figures you do not want to use generic labels like:

* Table 1
* Table 2

Instead, including explicit, descriptive labels, like:

* Table 1 – Income Statement (AUD millions, FY24)
* Table 2 – Balance Sheet (AUD millions, FY24)

## Machine-Readable Tables

Tables should always be formatted so that both humans **and** AI tools (e.g. NLP models, OCR extractors, Excel/Power BI scripts) can reliably read them. The goal is that **each row and column can stand alone without relying on formatting tricks such as merged cells, blank rows, or nested headers.**

Key Rules

* **No merged cells:** Each row should repeat its parent category (e.g. “Income Statement”) instead of merging cells vertically.
* **One header per column:** Avoid multiple header rows stacked on top of each other (e.g. “Financials” above “Revenue/EBITDA/NPAT”).
* **No spacer rows or blank columns:** Use white space or formatting (e.g. horizontal lines, shading) if you want to group data visually, but do not insert empty rows or columns.
* **Flat structure:** Every row should contain all the context needed (e.g. Area + Item + Value).

**Why This Matters**

* **For Humans:** Tables remain clean and can still be styled with visual separators (lines, bold headings) to aid readability.
* **For AI & Automation:** Flat tables avoid ambiguity, ensuring data is extracted accurately. For example:
  + OCR won’t “lose” the Income Statement label.
  + NLP models can directly map values to categories.
  + Data can be exported to CSV/JSON without needing manual clean-up.
* **For Regulators & Standards:** This approach is aligned with XBRL, APRA, and ISO reporting principles, which all prohibit merged cells in structured reporting.

**Key Principle:**

* Every table must be designed so that each row is self-contained and each column has a single, unambiguous header.

### Good vs Bad Examples

Bad (merged cells, nested headers, blank rows):

|  |  |
| --- | --- |
|  | FY24 (AUDm) |
| Income Statement | |
| Revenue | 100 |
| EBITDA | 45 |
| NPAT | 35 |

|  |  |
| --- | --- |
| Balance Sheet |  |
| Total Assets | 1000 |
| Liabilities | 800 |

Problems:

* “Income Statement” is merged across multiple rows.
* If extracted to CSV, rows for EBITDA/NPAT would lose their “Income Statement” label.
* AI/automation can misinterpret blank cells as “null” values.

Good (flat, machine-readable):

|  |  |  |
| --- | --- | --- |
| Area | Item | FY24 (AUDm) |
| Income Statement | Revenue | 100 |
| Income Statement | EBITDA | 45 |
| Income Statement | NPAT | 35 |
| Balance Sheet | Total Assets | 1000 |
| Balance Sheet | Liabilities | 800 |

Advantages:

* Each row contains all the required information.
* Easy to parse into Excel, Power BI, or Python (pandas).
* Consistent structure means AI can classify values without guessing.

## Inline Metadata

Every document should begin with a short block of **inline metadata** containing the key identifiers for the company and assessment. This metadata acts like a “cover page for machines,” ensuring that scripts, databases, and AI tools can immediately locate the critical facts without scanning through long paragraphs.

Metadata should be placed at the **very top of the document**, before the narrative sections, and follow a consistent key–value format.

**Key Rules**

* **Use explicit field labels:** Each item should have a fixed, descriptive label (e.g. Company Name: not Name:).
* **Keep it flat and simple:** One line per field, no merged cells or bullets.
* **Be consistent:** Always use the same field names across all documents (e.g. Rating: not sometimes Credit Rating:).
* **Cover core identifiers:** At a minimum include:
  + Company Name:
  + Industry:
  + Date:
  + Prepared By:
  + Document Version:
  + Internal Rating:

**Why This Matters**

* **For Humans:** Provides a clear front-matter summary, so readers don’t need to hunt through the report for key identifiers.
* **For AI & Automation:** Scripts can grab metadata quickly, reducing the risk of errors when pulling information into models, databases, or dashboards.
* **For Consistency:** Ensures reports are standardised across different authors and teams, avoiding ambiguity.
* **For Compliance:** Mirrors best practice in regulatory templates (e.g., APRA returns require top-level metadata fields before tabular data).

**Key Principle:**

*All reports must begin with a short block of inline metadata containing consistent, explicit labels for company identifiers, rating, author, and date.*

### Good vs Bad Examples

**Bad (implicit, inconsistent metadata):**

Company: Testa

Analyst: Tye

July 2025 report

BBB+ internal

Problems:

* Field labels are inconsistent (“Company” vs “Company\_Name”).
* Date not in a standard format.
* Rating is buried in free text.

**Good (explicit, inline metadata):**

Company Name: Testa Public Company Limited

Industry: Energy – Coal and Renewables

Date: 2025-07-31

Prepared By: Tye Fraser

Document Version: 1.0

Internal Rating: BBB+ (Stable)

Advantages:

* Each line is unambiguous and machine-readable.
* Consistent labels allow automated extraction across hundreds of documents.
* Humans can still read it at a glance.

## Separation of Narrative vs Data

When drafting reports, keep **narrative text** (explanations, commentary, analysis) clearly separated from **structured data** (tables, bullet lists, ratios). Mixing the two makes reports harder for both humans and AI to interpret.

**Key Rules**

* **Paragraphs = narrative:** Use full sentences for explanations, background, or qualitative commentary.
* **Tables/lists = data:** Use structured formats for numbers, ratios, and other fact sets.
* **Do not embed data inside narrative text:** Avoid writing long prose where key figures are hidden in sentences.
* **Do not mix commentary inside tables:** Tables should contain data only, not analysis or footnotes (except a clearly separated note below).

**Why This Matters**

* **For Humans:** Reports are cleaner and easier to skim — data blocks vs explanations are clearly distinguished.
* **For AI & Automation:** Structured data can be parsed without ambiguity, while narrative commentary can be processed separately for sentiment or keyword analysis.
* **For Consistency:** When reports are aggregated, data tables can be pulled directly into dashboards, while narratives stay as qualitative notes.

**Key Principle:**

*Keep commentary and narrative in prose, and keep data in tables or lists. Never mix the two in the same block of text.*

### Good vs Bad Examples

**Bad (mixed narrative and data):**

The company achieved revenue of 100m in FY24, an EBITDA margin of 45%, and NPAT of 35m, reflecting a solid result compared with prior years.

Problems:

* Data is embedded in narrative.
* AI may struggle to extract numbers without also pulling irrelevant text.
* Harder to build comparisons across reports.

**Better (separated narrative and data):**

**Narrative (paragraph):**

The company achieved strong financial performance in FY24, with growth across revenue, earnings, and net profit. Margins remained resilient despite higher input costs.

**Structured Data (table):**

|  |  |
| --- | --- |
| Metric | FY24 (AUDm) |
| Revenue | 100 |
| EBITDA | 45 |
| NPAT | 35 |

Advantages:

* Data is in a clean, machine-readable table.
* Narrative is still clear and readable.
* AI (or a parser) can reliably pull financials from the table, while still understanding the commentary separately.

**Bad (commentary inside a table):**

| **Metric** | **FY24 (AUDm)** |
| --- | --- |
| Revenue | 100 – up 10% from FY23 due to strong coal sales |
| EBITDA | 45 – margins improved after cost reductions |
| NPAT | 35 – supported by lower finance costs |

Problem:

* Data and narrative are interleaved.
* Parsing tools will read the “value” as text instead of a number.

**Better (separate note below):**

| Metric | FY24 (AUDm) |
| --- | --- |
| Revenue | 100 |
| EBITDA | 45 |
| NPAT | 35 |

*Note: Revenue growth was driven by stronger coal sales (+10%). EBITDA margins improved due to cost reduction initiatives.*

## Versioning

Every document should include clear **versioning metadata** so that both humans and AI can identify which iteration of the document they are working with. This avoids confusion when multiple drafts, updates, or resubmissions exist, and ensures automation can correctly capture the latest version.

**Key Rules**

* **Include version metadata at the top of the document** (recommended, alongside inline metadata).
* **Use a standard field name:** e.g. Document Version: not Version: in one file and Doc\_V: in another.
* **Use a consistent numbering format:**
  + Drafts → 0.1, 0.2, etc.
  + First release → 1.0
  + Minor edits → 1.1, 1.2
  + Major revisions → 2.0, 3.0
* **Always include author and date fields** with the version, so changes can be traced.
* **Optional:** Keep a version history table if documents undergo frequent revisions.

**Why This Matters**

* **For Humans:** Avoids confusion about which draft is the “final” version. Clear change tracking improves collaboration.
* **For AI & Automation:** Scripts can easily check Document Version to ensure they are ingesting the most recent file.
* **For Compliance:** Regulators and auditors often require documented version control — embedding this makes compliance straightforward.

📌 **Key Principle:**  
*All documents must include a version number, author, and date at the top. If documents are revised frequently, a version history table should be maintained.*

### Example: Inline Metadata with Versioning

| **Field** | **Value** |
| --- | --- |
| Company Name | Banpu Public Company Limited |
| Industry | Energy – Coal & Renewables |
| Date | 2025-07-31 |
| Prepared By | Tye Fraser |
| Document Version | 1.0 |
| Internal Rating | BBB+ (Stable) |

**Example: Version History Table**

| **Version** | **Date** | **Author** | **Notes** |
| --- | --- | --- | --- |
| 0.1 | 2025-07-20 | T. Fraser | First draft created |
| 0.2 | 2025-07-25 | T. Fraser | Added financials section |
| 1.0 | 2025-07-31 | T. Fraser | Finalised for distribution |
| 1.1 | 2025-08-15 | J. Smith | Updated with FY25 interim results |

## Font Choice for Machine Readability

The choice of font impacts how reliably AI or OCR tools can read text. Certain fonts make it difficult to distinguish similar characters (e.g. **capital “I” vs lowercase “l,”** or the letter **O** vs the digit **0**). Poor font selection can lead to extraction errors in automated systems.

**Key Rules**

* **Use sans-serif fonts** (e.g. Arial, Verdana, Tahoma, Calibri) for all body text.
  + Arial (i I l L 0 1),
  + Verdana (i I l L 0 1),
  + Tahoma (i I l L 0 1),
  + Calibri (i I l L 0 1)
* **Avoid decorative or cursive fonts**, which reduce recognition accuracy.
* **Use monospaced fonts** (e.g. Consolas, Courier New) when presenting tabular data or code, as each character has a fixed width.
* **For maximum clarity**, consider fonts designed for accessibility and OCR such as Atkinson Hyperlegible or OCR-B.
* **Maintain consistent font usage** throughout the document; do not mix multiple body fonts.
* **Ensure adequate font size** (minimum 11pt body text, 12pt or larger preferred for OCR accuracy).

**Why This Matters**

* **For AI & OCR:** Fonts with high character differentiation reduce misreads during parsing.
* **For Humans:** Clear fonts improve readability and accessibility for colleagues.
* **For Consistency:** Standardising fonts across all templates avoids errors when exporting to PDF, HTML, or structured formats.

📌 **Key Principle:**

*Always use simple, sans-serif or monospaced fonts that maximise the distinction between visually similar characters.*

### Good vs Bad Examples

**Bad (ambiguous font):**

Text written in decorative serif fonts where “1” (one) looks like “l” (ell) or “I” (capital i).

**Good (machine-friendly font):**

Text written in Arial or Verdana, where “1”, “l”, and “I” are visually distinct.

## Other notes

* Write elaborate content rather than brevity when writing instructional content. This ensures that a wide range of scenarios and topics are covered, and more information is available
* Create FAQs for each article
* Be receptivity specific, don’t just say “it was a great night” say “Financial Year end 2024 was a great night” – repeating the context of what is being referred to.

| **Action Items (11 Steps)** | **Why It Matters** |
| --- | --- |
| Use a clear, descriptive title | Helps AI and users identify your topic |
| Write short lead sections (intro) | Gives quick summary for context |
| Follow heading hierarchy (H1, H2, H3, etc.) | Organizes content for easy parsing |
| Start sections with a defining paragraph | Defines the focus for AI and readers |
| Break into clear sections and paragraphs | Improves scan-ability and processing |
| Add infoboxes or summary tables | Highlights key facts up front |
| Link only the first occurrence (internal) | Reduces clutter, strengthens signals |
| Use synonyms for main topics | Covers more search queries |
| Keep related words/entities close together | Boosts context for AI parsing |
| Write subject-predicate-object statements | Makes relationships clear |
| Clarify entities/definitions early | Avoids confusion, improves citations |

 **Consistent terminology.** To avoid confusing LLMs, use consistent business terms (e.g. always “Income Statement,” not sometimes “Profit & Loss”)[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content) and define key terms or include a glossary[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content).

 **Document metadata and versioning.** Including fields such as author, date and version number helps scripts quickly locate key information[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content).

 **Machine‑readable tables.** Tables should be flat—no merged cells, nested headers or blank spacer rows—so that each row stands on its own. Azure’s Document Intelligence service notes that high‑quality scans or source documents are easier to process[learn.microsoft.com](https://learn.microsoft.com/en-us/azure/ai-services/document-intelligence/prebuilt/layout#:~:text=Input%20requirements). Avoid scanning printed pages; instead, supply digital originals (DOCX, HTML or Markdown) whenever possible[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=LLMs%20have%20a%20much%20easier,structured%2C%20and%20semantically%20rich).

 **Accessible formatting for screen readers.** While not explicitly about AI, accessibility guidelines align with AI‑readiness: use Word’s built‑in heading styles rather than manually bolding text; define table header rows; and provide alternative text for images[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=When%20creating%20headings%2C%20it%E2%80%99s%20important,have%20a%20different%20visual%20appearance). This ensures that both AI models and assistive technologies correctly interpret structure.

1. *Recognizing letters in the image:* [**Optical character recognition**](https://www.ibm.com/think/topics/optical-character-recognition) is when a computer tries to extract text from a picture of text. Depending on the image quality and the font used, a computer may mistake some characters for others — such as a capital I for a lowercase L (“l”) or a letter O for a number zero (0).

A close-up of a paper

AI-generated content may be incorrect.

Don’t copy directly from Ai to your document:

<AI> 🡪 <AI editing document (perform checks) 🡪 <Final Document>

Note: We have to come up with standards on how to develop templates 🡪 then we can get AI to write templates for us!!

* **What the research says about AI‑readable content**
* **Use of clear headings and hierarchy.** Guidance from Document360 notes that content should have clear headings and subheadings and that bullet/numbered lists should be used to organize information[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content). Similarly, the Nava toolkit stresses that LLMs read content more easily when it is clear, well‑structured and semantically rich. It recommends frequent headings to describe sections, a clear hierarchy of headings and subheadings, and bullet or numbered lists rather than run‑on sentences[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=LLMs%20have%20a%20much%20easier,structured%2C%20and%20semantically%20rich).
* **Plain language and definitional first sentences.** Nava also encourages the use of plain language and concise definitions for key terms and recommends including document metadata such as dates and topic tags[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content). The Publisher Desk’s article on formatting content for AI search adds that each section should begin with a descriptive paragraph that clearly defines the section’s focus, and that headings should follow a strict hierarchy (H1 for major sections, H2 for subsections, etc.)[publisherdesk.com](https://www.publisherdesk.com/formatting-content-for-ai-search/#:~:text=Amazingly%2C%20I%E2%80%99ve%20read%20lots%20of,presented%20in%20a%20logical%20order).
* **Descriptive titles and summaries.** AI search guidelines advise using clear, descriptive titles and a short introductory section to summarize the document[publisherdesk.com](https://www.publisherdesk.com/formatting-content-for-ai-search/#:~:text=If%20you%E2%80%99re%20ready%20to%20adapt%2C,consider%20for%20your%20content%E2%80%99s%20formatting). This makes it easier for AI to identify the topic and extract relevant facts.
* **Consistent terminology.** To avoid confusing LLMs, use consistent business terms (e.g. always “Income Statement,” not sometimes “Profit & Loss”)[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content) and define key terms or include a glossary[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content).
* **Document metadata and versioning.** Including fields such as author, date and version number helps scripts quickly locate key information[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content).
* **Machine‑readable tables.** Tables should be flat—no merged cells, nested headers or blank spacer rows—so that each row stands on its own. Azure’s Document Intelligence service notes that high‑quality scans or source documents are easier to process[learn.microsoft.com](https://learn.microsoft.com/en-us/azure/ai-services/document-intelligence/prebuilt/layout#:~:text=Input%20requirements). Avoid scanning printed pages; instead, supply digital originals (DOCX, HTML or Markdown) whenever possible[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=LLMs%20have%20a%20much%20easier,structured%2C%20and%20semantically%20rich).
* **Accessible formatting for screen readers.** While not explicitly about AI, accessibility guidelines align with AI‑readiness: use Word’s built‑in heading styles rather than manually bolding text; define table header rows; and provide alternative text for images[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=When%20creating%20headings%2C%20it%E2%80%99s%20important,have%20a%20different%20visual%20appearance). This ensures that both AI models and assistive technologies correctly interpret structure.
* **Gaps and recommendations for your template**

1. **Replace Markdown‑style “#” headings with Word heading styles and numbered headings.** In a Word environment, hashtags (#) are not interpreted as headings; they appear as literal characters. Use Word’s Heading 1/Heading 2 styles and numbered sections (e.g. “1 Financials,” “1.1 Income Statement”) to create a consistent hierarchy. This aligns with ISO 2145 numbering and the hierarchy recommended in the AI‑readability guidelines[publisherdesk.com](https://www.publisherdesk.com/formatting-content-for-ai-search/#:~:text=Amazingly%2C%20I%E2%80%99ve%20read%20lots%20of,presented%20in%20a%20logical%20order).
2. **Make headings descriptive rather than generic.** Instead of “Document Metadata” or “Table 1,” include the actual subject: for example, “Financials – Income Statement (AUD millions)”[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content). Every table should have a descriptive caption.
3. **Flatten tables and avoid merged cells.** In the sample template, the top metadata table is already flat, but the financial sections could be misread if you decide to merge category cells. Use repeated values (e.g. “Income Statement” on every row) rather than merged cells to ensure each row is self‑contained. Do not leave blank rows or columns.
4. **Expand the metadata block.** Add Document Version, Prepared By, and possibly Department or Reviewer fields so that versioning is explicit and auditable[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content). Keep this metadata at the very top of the document (without numbering) so AI can retrieve it quickly.
5. **Include a version‑history table when documents undergo multiple revisions.** A simple table with columns such as “Version,” “Date,” “Author,” and “Notes” will help both colleagues and scripts track changes.
6. **Add a short summary/introduction.** According to AI‑search guidelines, a concise lead paragraph below the title summarizing the document’s purpose aids both AI and human readers[publisherdesk.com](https://www.publisherdesk.com/formatting-content-for-ai-search/#:~:text=If%20you%E2%80%99re%20ready%20to%20adapt%2C,consider%20for%20your%20content%E2%80%99s%20formatting).
7. **Use plain language and define key terms.** Write narrative sections in clear sentences, avoiding jargon, and provide definitions for technical terms or include a glossary[navapbc.com](https://www.navapbc.com/toolkits/readable-ai-content#:~:text=,tags%20that%20describe%20the%20content). This reduces ambiguity and improves AI comprehension.
8. **Use bullet/number lists for itemized content.** Instead of embedding lists in paragraphs, use bulleted or numbered lists to convey steps or elements[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content). This improves parsing.
9. **Provide alt text for any images, charts or diagrams.** Although your current template doesn’t include images, adding alt‑text descriptions ensures that AI models and screen readers can understand them.
10. **Standardize terminology across templates.** Ensure that all analysts use the same field names and section titles (“Income Statement,” “Balance Sheet,” “Cash Flow,” etc.) to reduce confusion[document360.com](https://document360.com/blog/technical-writing-ai-guidelines/#:~:text=information,related%20content).

* **Conclusion**

Your current template captures the basic structure of a credit assessment, but to make it truly AI‑ready you should standardize headings, provide explicit and descriptive labels, flatten tables, add versioning and metadata, and embrace clear hierarchical formatting using Word’s built‑in styles. Following the best practices outlined above will improve machine readability, support better information retrieval, and enhance the document’s accessibility and usability for both AI systems and human colleagues.