

Mickaël
CANOUIL

Mickaël CANOUIL Template

Complete Guide

Mickaël CANOUIL, Ph.D.^{1*} and Jane Doe²

Thursday, the 29th of January, 2026

Abstract

This document showcases all features of the Mickaël CANOUIL template for Quarto, spanning HTML, Typst (PDF), and Reveal.js (presentations) formats. It demonstrates template-specific components including value boxes, info panels, status badges, dividers, progress bars, cards, card grids, executive summaries, and highlighted blocks. Additionally, it covers standard Quarto features such as figures, tables, mathematical notation, callouts, code blocks, code annotations, mermaid diagrams, and cross-references. Use this document as both a learning resource and a quick reference guide for all available template capabilities.

Keywords: Quarto, Typst, Template, Documentation, Tutorial

Affiliations

¹ Independent Contractor, Lille, France

² Department of Statistics, Example University, London, United Kingdom

ORCID

Mickaël CANOUIL, Ph.D.:  [0000-0002-3396-4549](https://orcid.org/0000-0002-3396-4549)

Jane Doe:  [0000-0001-2345-6789](https://orcid.org/0000-0001-2345-6789)

*Corresponding author: pro@mickael.canouil.dev

Table of Contents

- 1. Introduction 9
 - 1.1. Document Organisation 10
- 2. Template-Specific Components 11
 - 2.1. Value Boxes 12
 - 2.1.1. Basic Value Boxes 12
 - 2.1.2. Value Box Icons 12
 - 2.1.3. Value Box Grid Layout 13
 - 2.2. Info Panels 13
 - 2.2.1. Panel Style Variants 13
 - 2.2.2. Status-Based Panels 14
 - 2.2.3. Panels with Rich Content 14
 - 2.3. Status Badges 15
 - 2.3.1. Badge Colour Types 15
 - 2.3.2. Badges with Icons 15
 - 2.3.3. Multiple Badges 15
 - 2.4. Dividers 15
 - 2.4.1. Basic Divider 16
 - 2.4.2. Divider with Label 16
 - 2.4.3. Divider Style Variants 16
 - 2.5. Progress Bars 16
 - 2.5.1. Progress Bar with Custom Colours 17
 - 2.6. Card Grids 17
 - 2.6.1. Basic Card Grid 17
 - 2.6.2. Card Style Variants 17
 - 2.6.3. Standalone Cards 18
 - 2.7. Executive Summary 18
- 3. Quarto Features 20
 - 3.1. Figures 21
 - 3.1.1. Simple Figure 21
 - 3.1.2. Figure with Custom Width 22
 - 3.1.3. Subfigures 22
 - 3.2. Tables 23
 - 3.2.1. Simple Table 23
 - 3.2.2. Tables with Subtables 23
 - 3.2.3. Table with Column Alignment 23
 - 3.2.4. Complex Data Table 23
 - 3.3. Mathematics 24
 - 3.3.1. Inline Mathematics 24
 - 3.3.2. Display Mathematics 24

Table of Contents

- 3.4. Callouts 24
- 3.5. Panel Tabsets 25
 - 3.5.1. Overview 25
 - 3.5.2. Details 25
 - 3.5.3. Examples 26
- 3.6. Code Blocks 26
 - 3.6.1. Inline Code 26
 - 3.6.2. Code Windows 26
 - 3.6.3. Python Code Block 27
 - 3.6.4. R Code Block 27
 - 3.6.5. Bash Script 28
- 3.7. Grid Background 28
- 3.8. Mermaid Diagrams 29
- 3.9. Code Annotations 29
- 4. Typography and Layout 30
 - 4.1. Heading Levels 31
 - 4.2. Level 3 Heading 31
 - 4.2.1. Level 4 Heading 31
 - 4.3. Lists 31
 - 4.3.1. Unordered Lists 31
 - 4.3.2. Ordered Lists 31
 - 4.3.3. Definition Lists 31
 - 4.4. Quotations and Asides 32
 - 4.4.1. Block Quotes 32
 - 4.4.2. Asides 32
 - 4.5. Layout Features 33
 - 4.5.1. Multi-Column Layout 33
 - 4.5.2. Three-Column Layout 33
 - 4.5.3. Highlighted Text Blocks 33
 - 4.6. Section Pages 33
 - 4.7. Cross-References 34
- 5. Advanced Features 35
 - 5.1. Footnotes 36
 - 5.2. Citations and Bibliography 36
 - 5.3. Custom Cross-References 36
 - 5.4. Document Types (Typst) 36
 - 5.5. Reveal.js Closing Slide 37
 - 5.6. Reveal.js Extension Options 37
 - 5.7. Special Sections 38
- 6. Conclusion 39
 - 6.1. Template-Specific Components 40
 - 6.2. Quarto Features 40
 - 6.3. Typography and Layout 40
 - 6.4. Advanced Features 41

- 6.5. Customisation Options 41
- 6.6. Accessibility 41
- 6.7. Getting Started 41
- References 42
- Appendix A. Additional Configuration 44
 - A.a. Configuration Examples 45
 - A.b. Complete YAML Options Reference 45
 - A.b.1. Style Options 45
 - A.b.2. Decorative Elements 46
 - A.b.3. Logo Configuration 46
 - A.b.4. Watermark Options 46
 - A.b.5. Typography 46
 - A.b.6. Page Layout 47
 - A.b.7. Document Structure 47
 - A.b.8. Accessibility 47
 - A.b.9. Special Sections 47
 - A.b.10. Table Styling 48
 - A.b.11. Links and Quotes 48
 - A.b.12. Professional Style Options 48
 - A.b.13. Extensions Configuration 48
 - A.b.14. Closing Slide (Reveal.js) 49
 - A.b.15. Margin Section Configuration 49
 - A.b.16. Default Values 50
 - A.b.17. Validation 50
 - A.c. Feature Reference Table 51
- Supplementary I. Colour Schemes 52
 - I.i. Light Mode Colours 53
 - I.ii. Dark Mode Colours 53
 - I.iii. Callout Colours (WCAG AA Compliant) 53

Downloaded from <http://ajph.org/> on November 10, 2014

List of Texts

Text 5.1 Example text demonstrating custom cross-reference types. 36

DEMO

List of Texts

1. Introduction

DEMO

1.1. Document Organisation

This document showcases all features of the Mickaël CANOUIL template for Quarto. It covers HTML, Typst (PDF), and Reveal.js (presentations) formats. It serves as both an educational resource and a quick reference guide.

Navigate using the table of contents to find specific features. Each section includes working examples you can copy and adapt for your own documents.

1.1. Document Organisation

This demonstration is organised into five main sections:

1. Template-Specific Components (Section 2): Value boxes, panels, badges, dividers, progress bars, card grids, executive summaries, grid background, mermaid diagrams, and code annotations.
2. Quarto Features (Section 3): Figures, tables, mathematics, callouts, tabsets, and code blocks.
3. Typography and Layout (Section 4): Headings, lists, quotations, asides, and layout features.
4. Advanced Features (Section 5): Footnotes, citations, custom cross-references, document types, Reveal.js options, and special sections.
5. Conclusion (Section 6): Summary of all features covered.

Cross-references work throughout this document. For example, see Section 2 for template components or Figure 3.1 for a figure example.

2. Template-Specific Components

2.1. Value Boxes

- 2.1.1. Basic Value Boxes
- 2.1.2. Value Box Icons
- 2.1.3. Value Box Grid Layout

2.2. Info Panels

- 2.2.1. Panel Style Variants
- 2.2.2. Status-Based Panels
- 2.2.3. Panels with Rich Content

2.3. Status Badges

- 2.3.1. Badge Colour Types
- 2.3.2. Badges with Icons
- 2.3.3. Multiple Badges

2.4. Dividers

- 2.4.1. Basic Divider
- 2.4.2. Divider with Label
- 2.4.3. Divider Style Variants

2.5. Progress Bars

- 2.5.1. Progress Bar with Custom Colours

2.6. Card Grids

- 2.6.1. Basic Card Grid
- 2.6.2. Card Style Variants
- 2.6.3. Standalone Cards

2.7. Executive Summary

The template provides nine specialised components for professional document creation. These components are unique to this Typst template and enhance visual communication.

2.1. Value Boxes

Value boxes display metrics, KPIs, and statistics in a visually appealing format. They are ideal for dashboards, reports, and presentations.

2.1.1. Basic Value Boxes

42 %
Growth Rate

1.2M €
Total Revenue

98 % ✓
Customer Satisfaction

24 
Active Projects

2.1.2. Value Box Icons

Value boxes support icons to indicate trends or status:

18 % ↑
Quarterly Growth

5 % ↓
Operating Costs

42% —
Market Share

2.1.3. Value Box Grid Layout

Arrange multiple value boxes using Quarto's `layout - ncol` feature:

42% ↑
Metric A

87% —
Metric B

15% ↓
Metric C

Value Box Parameters:

- `value` : The numeric value to display (required).
- `unit` : Unit symbol (% , € , \$, etc).
- `label` : Description text (required).
- `icon` : Icon shortcuts (up ↑ , down ↓ , stable —) or any UTF-8 character.
- `colour` : Colour types (success, warning, danger, info, neutral) or hex colours (#ff0000).
- `background` : Optional background colour override.
- `show-border` : Display border around the box (default: true).
- `alignment` : Content alignment (center, left, right; default: center).
- `alt` : Optional alt text for accessibility when using grid layouts (wraps grid in figure with alt text).

2.2. Info Panels

Info panels highlight content with eight distinct styles. They provide more flexibility than callouts and support rich content.

2.2.1. Panel Style Variants

Subtle panels provide minimal emphasis for background information or supplementary notes. Use when you want gentle visual separation without strong emphasis.

Emphasis panels offer moderate highlighting for important information that deserves attention. They create visual hierarchy without being overwhelming.

Accent panels create strong visual impact for key insights or critical information. Use sparingly for maximum effect.

Outline panels provide clean borders without background fill. They maintain clarity whilst keeping the design minimal and unobtrusive.

2.2.2. Status-Based Panels

i Information Panel

Info panels use note callout colours for explanatory content. Panel titles can be set via the `title` attribute or as the first heading inside the panel.

✓ Success Panel

Success panels highlight positive outcomes, achievements, or completed tasks. They use green tones to convey accomplishment and success.

⚠ Warning Panel

Warning panels alert readers to important caveats, limitations, or potential issues. Use them to draw attention to matters that require careful consideration.

— Critical Panel

Danger panels emphasise critical information, errors, or severe warnings. They demand immediate attention through bold red styling.

2.2.3. Panels with Rich Content

★ Rich Content Example

Panels support any content type:

- Unordered and ordered lists.
- Code blocks and inline code like `function()`.
- Mathematical notation: $E = mc^2$.
- Bold text, italic text, and bold italic text.

This versatility makes panels ideal for technical documentation and reports.

Panel Parameters:

- `style` : Style variant (subtle, emphasis, accent, outline, info, success, warning, danger).
- `title` : Optional title text (or use first heading inside panel).
- `icon` : Optional icon (any UTF-8 character or emoji).

- `border-width` : Border thickness (default: 1pt).
- `radius` : Corner radius (default: 8pt).
- `inset` : Internal padding (default: 1.2em).
- `breakable` : Allow page breaks inside panel (default: false).

2.3. Status Badges

Status badges provide compact inline indicators for status, categories, tags, and labels. They appear inline with text and use colour-coded styling.

2.3.1. Badge Colour Types

Project Status: Approved

Draft This section is under review.

Critical Requires immediate attention.

New Feature Recently added capability.

Internal For internal use only.

2.3.2. Badges with Icons

Badges support optional icons using any UTF-8 character or emoji:

✓ Completed ⌚ Pending ⚠ Error

i Beta 📄 Documentation

2.3.3. Multiple Badges

Combine multiple badges to provide comprehensive status information:

Document Status: Active High Priority Review Required

Badge Syntax:

- Span syntax (all formats): `[Text]{ .badge colour="success" icon="✓" }`.
- Shortcode (HTML and Reveal.js only):
`{ {< badge text="Text" colour="success" icon="✓" > } }`.

Badge Parameters:

- `colour` : Badge colour type (success, warning, danger, info, neutral).
- `icon` : Optional icon (any UTF-8 character or emoji).
- `inline` : Display inline or as block (default: true).

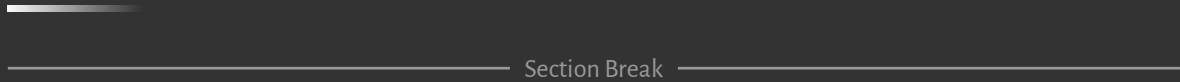
2.4. Dividers

Decorative dividers provide visual separation between sections without using headings.

2.4.1. Basic Divider



2.4.2. Divider with Label



2.4.3. Divider Style Variants



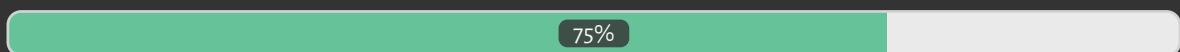
Divider Parameters:

- **style** : Divider style (solid, dashed, dotted, ornamental, gradient).
- **label** : Optional centred label text.
- **thickness** : Line thickness (default: 1pt).
- **width** : Line width as percentage (default: 50%).

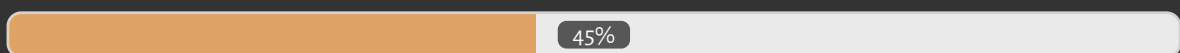
2.5. Progress Bars

Progress bars visualise completion, distribution, or proportions.

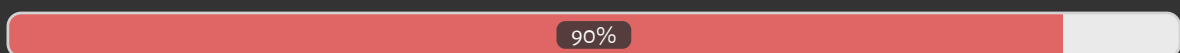
Project Completion



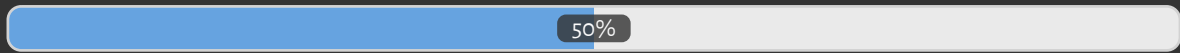
Budget Utilisation



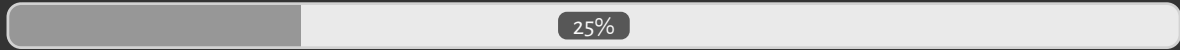
Server Capacity



Storage Used

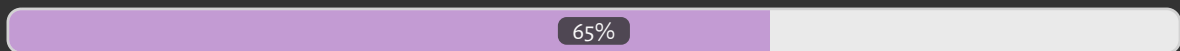


Baseline Progress



2.5.1. Progress Bar with Custom Colours

Custom Colour Progress



Progress Bar Parameters:

- `value` : Progress percentage (0-100, required).
- `label` : Description text (required).
- `colour` : Colour type (success, warning, danger, info, neutral) or hex colour.
- `height` : Bar height (default: 1.5em).
- `show-percentage` : Display percentage label (default: true).

2.6. Card Grids

Card grids provide flexible layouts for displaying content in card format.

2.6.1. Basic Card Grid

Performance

Optimised for speed with minimal overhead and efficient rendering across all platforms.

99.9% uptime guarantee

Security

Enterprise-grade security with end-to-end encryption, compliance, and regular audits.

SOC 2 Type II certified

Scalability

Scales effortlessly from small teams to global enterprises with millions of users.

10M+ active users

2.6.2. Card Style Variants

Cards support three distinct styles:

Subtle Style

Light background with subtle border. This is the default card style.

Outlined Style

Clear border with transparent background. Clean and minimal appearance.

Filled Style

Bold filled background with contrasting text. Maximum visual impact.

Card Grid Parameters:

- `columns` : Number of columns (1-4).
- `alt` : Optional alt text for accessibility (wraps grid in figure with alt text).
- Card `style` : subtle, outlined, or filled.
- Card `colour` : Custom accent colour (for filled style).
- Cards support title (`## Heading`), content, and footer (after `—`).

2.6.3. Standalone Cards

Individual cards can be used outside of card grids for single-item emphasis:

Standalone Card Example

This card is used independently, not within a card-grid. It provides the same styling options as cards within grids.

Filled Standalone Card

Filled cards with custom colours work perfectly as standalone components. Ideal for call-to-action blocks or featured content.

2.7. Executive Summary

Executive summary blocks provide prominent, professional summary sections for reports.

Executive Summary

This quarter exceeded all strategic targets across revenue, customer satisfaction, and project delivery.

Key Achievements:

- Revenue grew 18% year-over-year, surpassing projected targets.
- Customer satisfaction score reached 94%, a new company record.
- All major projects delivered on time and within budget.
- Team productivity improved 22% through process optimisation.

Strategic Priorities for Next Quarter:

1. Scale platform infrastructure to support 50,000 concurrent users.
2. Expand into three new European markets (Germany, France, Spain).
3. Enhance security framework with SOC 2 Type II certification.
4. Launch mobile application for iOS and Android platforms.

Overall Project Status: On Track High Confidence

Executive Summary Parameters:

- `title` : Custom title (default: "Executive Summary").
- `show-corner-brackets` : Include decorative corner brackets (default: true).

3. Quarto Features

3.1. Figures

- 3.1.1. Simple Figure
- 3.1.2. Figure with Custom Width
- 3.1.3. Subfigures

3.2. Tables

- 3.2.1. Simple Table
- 3.2.2. Tables with Subtables
- 3.2.3. Table with Column Alignment
- 3.2.4. Complex Data Table

3.3. Mathematics

- 3.3.1. Inline Mathematics
- 3.3.2. Display Mathematics

3.4. Callouts

3.5. Panel Tabsets

- 3.5.1. Overview
- 3.5.2. Details
- 3.5.3. Examples

3.6. Code Blocks

- 3.6.1. Inline Code
- 3.6.2. Code Windows
- 3.6.3. Python Code Block
- 3.6.4. R Code Block
- 3.6.5. Bash Script

3.7. Grid Background

3.8. Mermaid Diagrams

3.9. Code Annotations

The template supports all standard Quarto features with enhanced styling.

3.1. Figures

3.1.1. Simple Figure



Figure 3.1: An example figure demonstrating basic figure capabilities with caption and cross-referencing. Reference the figure using Figure 3.1 in your text.

3.1.2. Figure with Custom Width



Figure 3.2: A wider figure spanning more of the page width to demonstrate flexible sizing.
See Figure 3.2 for the wider figure example.

3.1.3. Subfigures



(a) First subfigure showing left panel.

(b) Second subfigure showing right panel.

Figure 3.3: Two subfigures arranged side by side demonstrating multi-panel layouts.

Reference the complete figure as Figure 3.3, or individual subfigures as Figure 3.3a and Figure 3.3b.

3.2. Tables

3.2.1. Simple Table

Table 3.1: A simple table demonstrating basic table capabilities.

Column A	Column B	Column C
Value 1	Value 2	Value 3
Value 4	Value 5	Value 6
Value 7	Value 8	Value 9

Reference tables using Table 3.1 in your text.

3.2.2. Tables with Subtables

Table 3.2: Two subtables arranged side by side demonstrating multi-table layouts.

(a) First subtable.		(b) Second subtable.	
Alpha	Beta	Gamma	Delta
A1	B1	G1	D1

3.2.3. Table with Column Alignment

Table 3.3: Column alignment demonstration (left, centre, right).

Left Aligned	Centre Aligned	Right Aligned
Text	Text	Text
123	456	789
Alpha	Beta	Gamma

See Table 3.3 for alignment examples.

3.2.4. Complex Data Table

Table 3.4: Template colour scheme comparison between light and dark modes.

Feature	Light Mode	Dark Mode	Notes
Background	#fafafa	#333333	Page background colour
Foreground	#333333	#fafafa	Primary text colour
Muted	#666666	#999999	Secondary text colour
Accent	#e67e22	#e67e22	Brand accent colour
Link	#0066cc	#66aa ff	Hyperlink colour
Corner Brackets	Enabled	Enabled	Signature branding element

Detailed colour information is available in Table 3.4.

3.3. Mathematics

3.3.1. Inline Mathematics

Inline mathematical expressions like $\alpha + \beta = \gamma$, $\int_0^1 f(x) dx$, and $\sum_{i=1}^n i = \frac{n(n+1)}{2}$ integrate seamlessly with text.

The quadratic formula $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ and Euler's identity $e^{i\pi} + 1 = 0$ demonstrate mathematical typesetting quality.

3.3.2. Display Mathematics

Einstein's famous mass-energy equivalence relation:

$$E = mc^2 \quad (1)$$

The quadratic formula for solving $ax^2 + bx + c = 0$:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (2)$$

Maxwell's equations governing electromagnetic fields:

$$\begin{aligned} \nabla \cdot \mathbf{E} &= \frac{\rho}{\epsilon_0} \\ \nabla \cdot \mathbf{B} &= 0 \\ \nabla \times \mathbf{E} &= -\frac{\partial \mathbf{B}}{\partial t} \\ \nabla \times \mathbf{B} &= \mu_0 \mathbf{J} + \mu_0 \epsilon_0 \frac{\partial \mathbf{E}}{\partial t} \end{aligned} \quad (3)$$

Matrix equation demonstrating linear algebra:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (4)$$

Cross-reference equations using Equation 1, Equation 2, Equation 3, or Equation 4.

3.4. Callouts

Quarto's standard callout blocks provide contextual information.

Note

Note callouts provide additional information or context without interrupting the main narrative flow. They are ideal for supplementary details that enhance understanding.

Pro Tip

Tip callouts offer helpful suggestions, best practices, or recommendations. Use them to share expertise and guide readers towards optimal approaches.

Important Information

Important callouts highlight essential details that readers must not miss. They emphasise critical information that affects understanding or outcomes.

Warning

Warning callouts alert readers to caveats, limitations, or potential issues. They help readers avoid common pitfalls or misunderstandings.

Caution

Caution callouts warn about significant risks, dangers, or problematic scenarios. They demand attention to prevent errors or adverse consequences.

3.5. Panel Tabsets

Panel tabsets organise content into tabs in HTML-based formats. In Typst (PDF), tabs are converted to a hierarchical heading structure since Typst does not support tabs.

3.5.1. Overview

This tab contains introductory content. In Typst, this becomes a subsection under “Overview” as a heading.

3.5.2. Details

This tab provides more detailed information.

- First detail item.
- Second detail item.
- Third detail item.

3.5.3. Examples

```
 PYTHON

# Example code in a tab
def greet(name: str) -> str:
    return f'Hello, {name}!'
```

Tabset Behaviour:

- HTML/Reveal.js: Renders as interactive tabs.
- Typst: Tab titles become headings; content headings are nested appropriately.

3.6. Code Blocks

3.6.1. Inline Code

Variable names like `my_variable`, function calls like `calculate_mean()`, and commands like `quarto render` appear inline with special formatting.

3.6.2. Code Windows

When `extensions.mcanouil.code-window.enabled: true` is set (the default), code blocks with a `filename` attribute display a macOS-style window header. With `extensions.mcanouil.code-window.auto-filename: true` (the default), code blocks without explicit filenames automatically show the language name in small-caps styling.

```
 fibonacci.py

def fibonacci(n: int) -> int:
    """Calculate the nth Fibonacci number."""
    if n <= 1:
        return n
    return fibonacci(n - 1) + fibonacci(n - 2)
```

```
 analysis.R

# Load data and create summary
data <- read.csv("data.csv")
summary(data)
```

With `auto-filename: true`, code blocks without explicit filenames show the language name:

3.6.3. Python Code Block

```
PYTHON

def fibonacci(n: int) -> int:
    """Calculate the nth Fibonacci number using recursion."""
    if n <= 1:
        return n
    return fibonacci(n - 1) + fibonacci(n - 2)

def fibonacci_iterative(n: int) -> int:
    """Calculate the nth Fibonacci number iteratively."""
    if n <= 1:
        return n
    a, b = 0, 1
    for _ in range(2, n + 1):
        a, b = b, a + b
    return b

# Example usage
result = fibonacci_iterative(10)
print(f"F(10) = {result}")
```

3.6.4. R Code Block

```
R

# Load required libraries
library(ggplot2)
library(dplyr)

# Create sample data
data <- data.frame(
  x = 1:10,
  y = c(2, 4, 6, 8, 10, 12, 14, 16, 18, 20)
)

# Create visualisation
ggplot(data, aes(x = x, y = y)) +
  geom_point(size = 3, colour = "steelblue") +
  geom_line(colour = "steelblue") +
  labs(
    title = "Linear Relationship",
    x = "X Variable",
    y = "Y Variable"
  ) +
  theme_minimal()
```

3.6.5. Bash Script

```
#!/bin/bash
# Automated backup script

SOURCE="/home/user/documents"
BACKUP="/backup/documents"
TIMESTAMP=$(date +%Y%m%d_%H%M%S)

# Create backup directory if it doesn't exist
mkdir -p "${BACKUP}"

# Compress and archive
tar -czf "${BACKUP}/backup_${TIMESTAMP}.tar.gz" "${SOURCE}"

# Verify backup
if [ $? -eq 0 ]; then
    echo "Backup completed successfully: backup_${TIMESTAMP}.tar.gz"
else
    echo "Backup failed" >&2
    exit 1
fi
```

3.7. Grid Background

The template supports a subtle grid background overlay for HTML and Reveal.js formats. This is controlled via the `extensions.mcanouil.grid-background` option.

YAML Configuration:

```
extensions:
  mcanouil:
    grid-background: true # Enable grid background (default: false)
```

When enabled, a faint grid pattern appears behind all content. The grid uses a 50px spacing with a colour derived from the foreground and background colours.

Format Availability:

- HTML: Applied to the page body.
- Reveal.js: Applied to slide backgrounds via a pseudo-element overlay.
- Typst: Not available (PDF format).

3.8. Mermaid Diagrams

The template supports Mermaid diagrams with automatic dark theme styling.

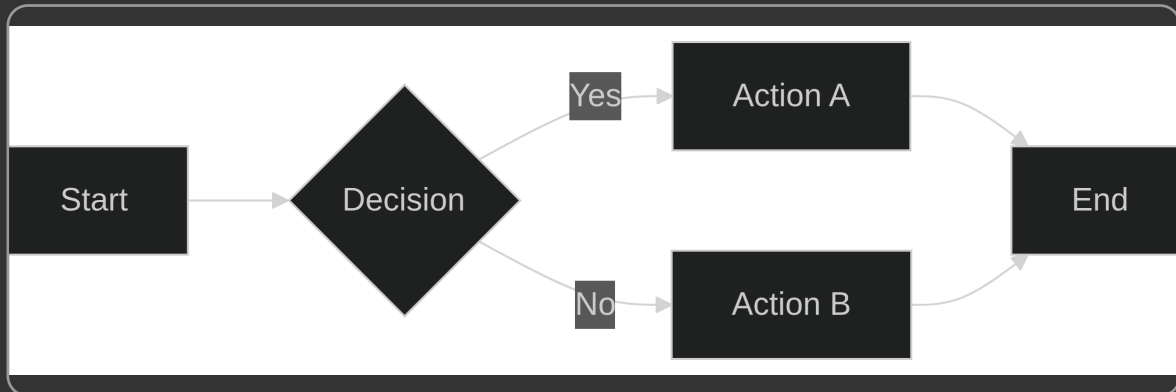


Figure 3.4: A simple flowchart demonstrating Mermaid diagram support.

See Figure 3.4 for the rendered diagram.

3.9. Code Annotations

Code annotations allow readers to explore code explanations interactively. In Reveal.js, the `code-annotation-fragments` plugin enables fragment-based navigation through annotations.

```
def calculate_area(radius: float) -> float: # <1>
    """Calculate the area of a circle.""" # <2>
    import math # <3>
    return math.pi * radius ** 2 # <4>
```

1. Function signature with type hints for input and output.
2. Docstring describing the function purpose.
3. Import the `math` module for access to `pi`.
4. Apply the area formula: $A = \pi * r^2$.

Configuration:

```
# Enable code annotations (set in YAML front matter)
code-annotations: select # Options: select, hover, none

# Reveal.js fragment navigation for annotations
extensions:
  mcanouil:
    code-annotation-fragments: true
```

4. Typography and Layout

DEMO

- 4.1. Heading Levels
- 4.2. Level 3 Heading
 - 4.2.1. Level 4 Heading
- 4.3. Lists
 - 4.3.1. Unordered Lists
 - 4.3.2. Ordered Lists
 - 4.3.3. Definition Lists
- 4.4. Quotations and Asides
 - 4.4.1. Block Quotes
 - 4.4.2. Asides
- 4.5. Layout Features
 - 4.5.1. Multi-Column Layout
 - 4.5.2. Three-Column Layout
 - 4.5.3. Highlighted Text Blocks
- 4.6. Section Pages
- 4.7. Cross-References

4.1. Heading Levels

This section demonstrates all six heading levels with their respective styling.

4.2. Level 3 Heading

Content under level 3 heading demonstrating hierarchy and spacing.

4.2.1. Level 4 Heading

Content under level 4 heading with appropriate font sizing.

4.2.1.1. Level 5 Heading

Content under level 5 heading maintaining visual consistency.

4.2.1.1.1. Level 6 Heading

Content under level 6 heading (the deepest available level).

4.3. Lists

4.3.1. Unordered Lists

- First item demonstrating unordered list styling.
- Second item with nested elements:
 - Nested item 2.1 showing second-level indentation.
 - Nested item 2.2 with consistent formatting.
 - Deeply nested item 2.2.1 demonstrating third-level nesting.
 - Deeply nested item 2.2.2 with proper alignment.
- Third item returning to top level.

4.3.2. Ordered Lists

1. First numbered item establishing sequence.
2. Second numbered item with sub-steps:
 - a. Sub-step 2.1 using alphabetic numbering.
 - b. Sub-step 2.2 maintaining hierarchy.
 - c. Sub-step 2.3 showing consistent styling.
3. Third numbered item completing the sequence.

4.3.3. Definition Lists

Term One

Definition of term one providing clear explanation and context.

Term Two

Definition of term two with more detailed information. Definitions can span multiple lines whilst maintaining proper formatting and indentation.

Complex Term

A more complex definition that includes bold text, italic text, and `inline code` to demonstrate rich formatting within definition lists.

4.4. Quotations and Asides

4.4.1. Block Quotes

“

“The best way to predict the future is to invent it.”

— Alan Kay

”

“

Longer block quotes demonstrate styling across multiple lines and paragraphs. They maintain consistent formatting, proper indentation, and visual hierarchy.

Multi-paragraph quotes preserve spacing and readability whilst clearly distinguishing quoted content from the main narrative.

”

4.4.2. Asides

⚠ Caution

`.aside` and `.column-margin` are not implemented in Typst yet.

This is regular paragraph text in the main column.

This is an aside note appearing in the margin. Asides provide supplementary information without interrupting the main flow.

More regular text continues in the main column after the aside.

Margin notes offer another way to include supplementary content. They appear in the page margin and support rich formatting.

4.5. Layout Features

4.5.1. Multi-Column Layout

Column 1: Content in the first column demonstrating side-by-side layout capabilities. This column contains the first half of the content with proper formatting and spacing.

Column 2: Content in the second column showing parallel information. This column contains the second half, creating a balanced two-column presentation.

4.5.2. Three-Column Layout

First: Brief content in column one.

Second: Brief content in column two.

Third: Brief content in column three.

4.5.3. Highlighted Text Blocks

This block uses highlighted styling for emphasis and visual impact. Use highlighted blocks for key takeaways, important statements, or critical information that deserves special attention. They create visual hierarchy whilst maintaining readability.

4.6. Section Pages

When `section-page: true` is enabled, level-1 headings are rendered on dedicated section pages with special styling:

- Banner: A coloured banner at 20% from the top of the page, spanning 75% of the page width from the left edge.
- Inverted colours: The banner uses the foreground colour as background, with the section title in the background colour.
- Heading underline: The banner includes the signature gradient underline with inverted colours.
- Subsection outline: Below the banner, an outline of subsections within that section is displayed (respecting `toc-depth`).
- Symmetric margins: Section pages use symmetric margins, similar to the title page.
- Single-column: Section pages are always single-column, regardless of the document's column setting.
- No margin section: The professional style margin section is suppressed on section pages.

Section pages apply to all level-1 headings except those with `outlined: false` (such as TOC and list-of sections).

YAML Configuration:

```
format:
  mcanouil-typst:
    section-page: true      # Enable section pages
    toc-depth: 3           # Controls outline depth on section pages
```

4.7. Cross-References

This document includes comprehensive cross-references throughout:

Sections:

- Introduction: Section 1.
- Template Components: Section 2.
- Value Boxes: Section 2.1.
- Grid Background: Section 3.7.
- Mermaid Diagrams: Section 3.8.
- Code Annotations: Section 3.9.
- Section Pages: Section 4.6.
- Figures: Section 3.1.
- Tables: Section 3.2.
- Mathematics: Section 3.3.
- Callouts: Section 3.4.
- Document Types: Section 5.4.
- Reveal.js Options: Section 5.6.
- Closing Slide: Section 5.5.

Figures:

- Simple figure: Figure 3.1.
- Wide figure: Figure 3.2.
- Mermaid diagram: Figure 3.4.
- Multiple subfigures: Figure 3.3 (specifically Figure 3.3a and Figure 3.3b).

Tables:

- Simple table: Table 3.1.
- Aligned table: Table 3.3.
- Colour scheme table: Table 3.4.

Equations:

- Einstein's equation: Equation 1.
- Quadratic formula: Equation 2.
- Maxwell's equations: Equation 3.
- Matrix equation: Equation 4.

5. Advanced Features

DEMO

- 5.1. Footnotes
- 5.2. Citations and Bibliography
- 5.3. Custom Cross-References
- 5.4. Document Types (Typst)
- 5.5. Reveal.js Closing Slide
- 5.6. Reveal.js Extension Options
- 5.7. Special Sections

5.1. Footnotes

This sentence includes a footnote.¹ Another sentence demonstrates multiple footnotes working together.²

Extended discussion with several footnotes creates a rich layer of supplementary information.³

5.2. Citations and Bibliography

Citations integrate seamlessly with the bibliography system. See Example and Author (2024) for a comprehensive example of citation formatting.

Multiple citations can be grouped together (Example and Author 2024, 2024), and citations work with inline text as well. According to Example and Author (2024), proper citation practices enhance academic rigour.

The bibliography appears automatically at the end of the document.

5.3. Custom Cross-References

The template supports custom cross-reference types beyond the standard sections, figures, tables, and equations.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis sagittis posuere ligula sit amet lacinia. Duis dignissim pellentesque magna, rhoncus congue sapien finibus mollis. Ut eu sem laoreet, vehicula ipsum in, convallis erat. Vestibulum magna sem, blandit pulvinar augue sit amet, auctor malesuada sapien. Nullam faucibus leo eget eros hendrerit, non laoreet ipsum lacinia. Curabitur cursus diam elit, non tempus ante volutpat a. Quisque hendrerit blandit purus non fringilla. Integer sit amet elit viverra ante dapibus semper. Vestibulum viverra rutrum enim, at luctus enim posuere eu. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.

Text 5.1: Example text demonstrating custom cross-reference types.

Reference custom floats using Text 5.1 in your text.

5.4. Document Types (Typst)

The Typst format supports multiple document types via the `document-type` parameter. Each type provides a distinct layout and structure suited to its purpose.

Available Document Types:

- `report` (default): Standard report layout with title page, table of contents, and section pages.
- `invoice`: Invoice layout for billing documents.
- `letter`: Formal letter layout.

¹First footnote providing additional context, citations, or explanatory information that supplements the main text without disrupting narrative flow.

²Second footnote demonstrating how multiple footnotes coexist within the same document whilst maintaining clear numbering and proper formatting.

³Third footnote showing that footnotes can contain formatting, styles, and even `code` to enhance their informativeness and utility.

- `cv` : Curriculum vitae layout.

YAML Configuration:

```
format:
  mcanouil-typst:
    document-type: report # Options: report, invoice, letter, cv
```

5.5. Reveal.js Closing Slide

The Reveal.js format supports a custom closing slide with a photo and social links.

YAML Configuration:

```
format:
  mcanouil-revealjs:
    closing-slide:
      photo: "https://github.com/mcanouil.png"
      social:
        website: https://mickael.canouil.fr
        github: mcanouil
        linkedin: MickaelCanouil
        bluesky: mickael.canouil.fr
        mastodon: https://fosstodon.org/@MickaelCanouil
```

The closing slide displays the author photo alongside social media links, providing a professional end to presentations.

5.6. Reveal.js Extension Options

The Reveal.js format supports several extension-specific options under the `extensions.mcanouil` namespace:

```
extensions:
  mcanouil:
    section-outline: true # Show subsection outline on section
slides.
  date-superscript: true # Format ordinal dates (1st, 2nd, 3rd).
  hide-title-slide-chrome: true # Hide menu/logo/footer on title slide.
  favicon-from-logo: true # Generate favicon from slide logo.
  code-annotation-fragments: true # Enable annotation fragment navigation.
  debug-borders: false # Show debug borders on slides.
```

5.7. Special Sections

Special sections (appendix and supplementary) automatically relocate to the document end with customised numbering.

6. Conclusion

DEMO

- 6.1. Template-Specific Components
- 6.2. Quarto Features
- 6.3. Typography and Layout
- 6.4. Advanced Features
- 6.5. Customisation Options
- 6.6. Accessibility
- 6.7. Getting Started

This demonstration showcases the complete feature set of the Mickaël CANOUIL template across HTML, Typst (PDF), and Reveal.js (presentations) formats.

6.1. Template-Specific Components

The template provides specialised components:

1. Value Boxes (Section 2.1): Display metrics, KPIs, and statistics with icons and colours.
2. Info Panels (Section 2.2): Highlight content with eight style variants (subtle, emphasis, accent, outline, info, success, warning, danger).
3. Status Badges (Section 2.3): Provide compact inline status indicators with five colour types.
4. Dividers (Section 2.4): Create visual separation with multiple style options (solid, dashed, dotted, gradient, ornamental).
5. Progress Bars (Section 2.5): Visualise completion and proportions with customisable colours.
6. Cards (Section 2.6): Individual card components for featured content with three style variants.
7. Card Grids (Section 2.6): Present multiple cards in flexible grid layouts.
8. Executive Summaries (Section 2.7): Create prominent summary blocks for professional reports.
9. Highlighted Blocks (Section 4.5): Draw attention to key information with subtle background emphasis.
10. Grid Background (Section 3.7): Subtle grid overlay for HTML and Reveal.js formats.
11. Code Annotations (Section 3.9): Interactive code explanations with fragment navigation.
12. Mermaid Diagrams (Section 3.8): Flowcharts and diagrams with automatic dark theme styling.

6.2. Quarto Features

Standard Quarto features work seamlessly with enhanced styling:

- Figures (Section 3.1): Simple figures, custom widths, subfigures, and featured image styling.
- Tables (Section 3.2): Simple tables, column alignment, and complex data tables.
- Mathematics (Section 3.3): Inline expressions and display equations with cross-references.
- Callouts (Section 3.4): Five callout types (note, tip, important, warning, caution).
- Tabsets (Section 3.5): Panel tabsets with automatic Typst conversion to headings.
- Code Blocks (Section 3.6): Syntax highlighting for Python, R, Bash, inline code, and macOS-style code windows.

6.3. Typography and Layout

Comprehensive typography and layout features:

- Headings (Section 4.1): Six heading levels with gradient underlines.
- Lists (Section 4.3): Unordered, ordered, and definition lists with nesting.
- Quotations (Section 4.4): Block quotes and asides with proper styling.
- Layout Features (Section 4.5): Multi-column layouts and highlighted blocks.
- Cross-References (Section 4.7): Seamless referencing throughout the document.

6.4. Advanced Features

- Document Types (Section 5.4): Report, invoice, letter, and CV layouts for Typst.
- Closing Slide (Section 5.5): Custom closing slide with photo and social links for Reveal.js.
- Reveal.js Options (Section 5.6): Section outlines, date formatting, and more.
- Special Sections (Section 5.7): Auto-relocating appendix and supplementary sections.

6.5. Customisation Options

The template supports extensive customisation through YAML parameters:

- Colour Modes: Light and dark modes with optional colour overrides.
- Decorative Elements: Corner brackets, margin decoration, title page backgrounds, and heading underlines.
- Typography: Custom fonts, sizes, weights, and line heights.
- Page Layout: Flexible paper sizes, margins, and column configurations.
- Watermarks: Text or image watermarks with customisable opacity, angle, and colour.
- Logo: Flexible logo positioning and sizing with alternative text.
- Special Sections: Configurable appendix and supplementary sections with custom numbering.

6.6. Accessibility

The template is designed with PDF accessibility (PDF/UA-1) compliance in mind:

- PDF/UA-1 Standard: Enabled by default via `pdf-standard: ua-1` (requires Typst ≥ 0.14).
- Semantic Structure: Proper heading hierarchy, tables with headers, and semantic elements.
- Decorative Elements as Artifacts: Corner brackets, margin decorations, watermarks, and other decorative elements are marked as PDF artifacts so screen readers skip them.
- Alt Text Support: Grid-based components (card grids, value box grids) support an optional `alt` parameter for accessibility descriptions.
- Non-Colour Differentiation: Callouts include default titles (Note, Tip, Warning, etc.) to ensure information is not conveyed by colour alone.
- WCAG AA Contrast: Colours are validated to meet WCAG AA contrast requirements.

6.7. Getting Started

To use these features in your own documents:

1. Install the template extension: `quarto add mcanouil/quarto-mcanouil@0.9.0`.
2. Set the format in YAML front matter: `format: mcanouil-html`, `format: mcanouil-typst`, or `format: mcanouil-revealjs`.
3. Customise template parameters as demonstrated in this document's YAML.
4. Copy and adapt examples from each section as needed.

References

DEMO

Example, Jane, and John Author. 2024. 'An Example Article for Demonstrating Citations'. Journal of Examples, ahead of print. <https://doi.org/10.1234/example.2024>.

DEMO

References

Appendix A. Additional Configuration

- A.a. Configuration Examples
- A.b. Complete YAML Options Reference
 - A.b.1. Style Options
 - A.b.2. Decorative Elements
 - A.b.3. Logo Configuration
 - A.b.4. Watermark Options
 - A.b.5. Typography
 - A.b.6. Page Layout
 - A.b.7. Document Structure
 - A.b.8. Accessibility
 - A.b.9. Special Sections
 - A.b.10. Table Styling
 - A.b.11. Links and Quotes
 - A.b.12. Professional Style Options
 - A.b.13. Extensions Configuration
 - A.b.14. Closing Slide (Reveal.js)
 - A.b.15. Margin Section Configuration
 - A.b.16. Default Values
 - A.b.17. Validation
- A.c. Feature Reference Table

This appendix section demonstrates the special sections feature. Content marked with the `.appendix` class automatically relocates to the document end with special numbering (A.1, A.2, etc).

A.a. Configuration Examples

Additional YAML configuration examples for advanced customisation:

```
format:
  mcanouil-typst:
    # Advanced Typography
    heading-colour: "#e67e22"           # Custom heading colour

    # Advanced Table Styling
    table-stroke: 1pt                   # Custom border thickness
    table-inset: 8pt                    # Custom cell padding
    table-fill: alternating             # Striped table rows

    # Advanced Quote Styling
    quote-width: 85%                   # Custom blockquote width
    quote-align: left                  # Custom blockquote alignment

    # Advanced Link Styling
    link-underline: false               # Disable link underlines
    link-colour: "#3498db"             # Custom link colour
```

A.b. Complete YAML Options Reference

This section documents all available YAML front matter options for customising the template.

A.b.1. Style Options

```
# Document style
style: "professional" # "academic" or "professional"

# Colour mode
brand-mode: "light" # "light" or "dark"

# Document type (Typst only)
document-type: "report" # "report", "invoice", "letter", or "cv"

# Override default colours (optional)
colour-background: "#FFFFFF" # Hex colour code
colour-foreground: "#333333" # Hex colour code
colour-muted: "#999999" # Hex colour code
```

A.b.2. Decorative Elements

```
● ● ● YAML

# Control decorative features
show-corner-brackets: true      # Corner brackets around content blocks
show-margin-decoration: true    # Decorative elements in margins
show-title-page-background: true # Geometric background on title page
show-heading-underlines: true   # Underlines beneath headings
```

A.b.3. Logo Configuration

```
● ● ● YAML

show-logo: true      # Display logo in header
logo-width: 40pt     # Logo width (omit to use height only)
logo-inset: 4pt      # Padding around logo
logo-alt: "Logo text" # Alternative text for accessibility
```

A.b.4. Watermark Options

```
● ● ● YAML

watermark-text: "DRAFT"      # Text watermark
watermark-image: "path/to/img" # Image watermark path
watermark-opacity: 10%       # Transparency (0-100%)
watermark-angle: "-45deg"    # Rotation angle
watermark-size: 4em          # Text size for text watermarks
watermark-colour: "gray"     # Colour for text watermarks
```

A.b.5. Typography

```
● ● ● YAML

# Fonts
font-body: "Alegreya Sans" # Body text font
font-headings: "Alegreya Sans" # Heading font
font-size: 11pt           # Base font size

# Heading styling
heading-weight: "bold"     # Font weight
heading-style: "normal"    # Font style (normal/italic)
heading-colour: null       # Override colour (null uses foreground)
heading-line-height: 0.65em # Line height for headings
title-size: 24pt          # Title font size
subtitle-size: 14pt        # Subtitle font size
```

A.b.6. Page Layout

```
papersize: "a4"      # Paper size (a4, letter, a5, etc.)
columns: 1           # Number of columns (1 or 2)

# Custom margins (optional)
margin:
  top: 2.5cm
  bottom: 2.5cm
  left: 2.5cm
  right: 2.5cm
```

A.b.7. Document Structure

```
toc: true            # Include table of contents
list-of: true        # Include list of figures/tables
number-sections: true # Number section headings
section-numbering: "1.1.1." # Numbering pattern
section-pagebreak: true # Page break before level 1 headings
section-page: true   # Dedicated section pages for level-1 headings
```

A.b.8. Accessibility

```
# PDF/UA-1 accessibility standard (requires Typst >= 0.14)
pdf-standard: ua-1 # Produces accessible tagged PDFs
```

A.b.9. Special Sections

```
special-sections:
  references:
    name: "References" # Section name
    numbering: "none" # Numbering pattern ("none" for unnumbered)
    prefix: false # Show section name prefix
  appendix:
    name: "Appendix"
    numbering: "A.a." # Appendix numbering (A.1, A.2, etc.)
    prefix: true # Show "Appendix" prefix
  supplementary:
    name: "Supplementary"
```

```
numbering: "I.1."      # Roman numerals (I.1, I.2, etc.)
prefix: true
```

A.b.10. Table Styling

```
table-stroke: "0.5pt"  # Border thickness
table-inset: 5pt       # Cell padding
table-fill: null       # Header background colour (null for foreground)
```

A.b.11. Links and Quotes

```
quote-width: 75%      # Blockquote width
quote-align: "center"  # Blockquote alignment
link-underline: true   # Underline hyperlinks
link-underline-opacity: 60% # Link underline transparency
link-colour: null      # Override link colour (null uses foreground)
```

A.b.12. Professional Style Options

Additional options available when `style: "professional"`:

```
institute: "Company Name" # Footer left content (defaults to first author
affiliation)
copyright: "© 2026 Company" # Footer right content
license: "MIT License"      # Licence info (combined with copyright if both
present)
title-page: true            # Dedicated title page mode
```

A.b.13. Extensions Configuration

```
extensions:
  mcanouil:
    # Grid background (HTML and Reveal.js)
    grid-background: false # Subtle grid overlay (default: false)
    # Code window styling (all formats)
    code-window:
      enabled: true          # macOS-style code windows (default: true)
      auto-filename: true    # Auto-show language name (default: true)
```

```
# HTML only
hide-navbar-title: true           # Hide title from navbar (default: true)
# Reveal.js only
section-outline: true            # Subsection outline on section slides
date-superscript: true           # Ordinal date formatting (1st, 2nd)
hide-title-slide-chrome: true    # Hide chrome on title slide
favicon-from-logo: true          # Generate favicon from logo
code-annotation-fragments: true  # Fragment navigation for annotations
debug-borders: false            # Debug borders on slides
```

A.b.14. Closing Slide (Reveal.js)

```
closing-slide:
  photo: "https://example.com/photo.png"
  social:
    website: https://example.com
    github: username
    linkedin: username
    bluesky: handle.bsky.social
    mastodon: https://instance/@user
```

A.b.15. Margin Section Configuration

Note: Margin sections are only displayed in professional style. Currently, margin section styling is controlled by internal constants in `margin-section.typ` :

- `MARGIN-SECTION-REFERENCE-SIZE` : Reference font size for measuring text widths (default: 12pt).
- `MARGIN-SECTION-FILL-RATIO` : Target fill ratio for vertical constraint; the longest heading occupies at most this percentage of available height (default: 90%).
- `MARGIN-SECTION-MAX-MARGIN-RATIO` : Maximum font size as percentage of left margin width after rotation (default: 50%).
- `MARGIN-SECTION-POSITION-HORIZONTAL` : Horizontal position percentage within margin (default: 85%).
- `MARGIN-SECTION-OPACITY` : Text opacity level (default: 70%).
- `MARGIN-SECTION-WEIGHT` : Font weight (default: "bold").
- `MARGIN-SECTION-ROTATION` : Rotation angle (default: -90deg for vertical text).
- `MARGIN-SECTION-MIN-SIZE` : Minimum text size for accessibility (default: 9pt).
- `MARGIN-SECTION-HEADING-LEVEL` : Heading level to display (default: 1 for top-level sections).

Behaviour:

- The margin section shows the most recent heading of the configured level.
- Section titles persist across pages until a new heading of that level appears.
- Supports special section numbering for appendices, supplementary materials, and references.

- Automatic font size calculation: Queries all headings in the document, measures their text width, and calculates the optimal font size subject to two constraints: (1) the longest heading fills at most 90% of available height, and (2) font size does not exceed 50% of the left margin width.
- Consistent sizing across all pages without text wrapping.
- Gracefully fails (returns none) if required parameters are missing.

Configuration options may be added in future versions for user customisation via YAML.

A.b.16. Default Values

If an option is not specified, the following defaults apply:

Option	Default	Description
<code>style</code>	<code>"academic"</code>	Academic style with simple header/footer
<code>document-type</code>	<code>"report"</code>	Document layout type (Typst only)
<code>brand-mode</code>	<code>"light"</code>	Light colour scheme
<code>show-corner-brackets</code>	<code>true</code>	Decorative brackets enabled
<code>show-logo</code>	<code>true</code>	Logo displayed when provided
<code>font-size</code>	<code>11pt</code>	Standard readable size
<code>columns</code>	<code>1</code>	Single-column layout
<code>section-pagebreak</code>	<code>true</code>	Level 1 headings start new page
<code>section-page</code>	<code>false</code>	Dedicated section pages disabled
<code>toc</code>	<code>false</code>	Table of contents not generated
<code>list-of</code>	<code>true</code>	List of figures/tables generated
<code>number-sections</code>	<code>true</code>	Section headings numbered
<code>pdf-standard</code>	<code>ua-1</code>	PDF/UA-1 accessibility standard

A.b.17. Validation

The template validates options and provides helpful error messages:

- Colour values must be valid hex codes or colour names.
- Size values must include units (pt, em, cm, etc.).
- Boolean values must be `true` or `false`.
- Special section numbering must be valid Typst numbering patterns.

A.c. Feature Reference Table

Table A.5: Complete feature reference with syntax and parameters.

Component	Shortcode/Class	Key Parameters
Value Box		value, unit, label, icon, colour, alt
Panel	.panel	style, title, icon
Badge	.badge	colour, icon
Divider		style, label, width, thickness
Progress Bar		value, label, colour
Card	.card	style, colour
Card Grid	.card-grid	columns, alt
Executive Summary	.executive-summary	title
Highlight	.highlight	(none)

Supplementary I. Colour Schemes

DEMO

I.i. Light Mode Colours

I.ii. Dark Mode Colours

I.iii. Callout Colours (WCAG AA Compliant)

This supplementary section demonstrates additional special section functionality. Content marked with `.supplementary` appears after appendices with Roman numeral numbering (I.1, I.2, etc).

I.i. Light Mode Colours

Table I.6: Light mode colour scheme.

Element	Hex Code	RGB	Usage
Background	#fafafa	rgb(250,250,250)	Page background
Foreground	#333333	rgb(51,51,51)	Primary text
Muted	#666666	rgb(102,102,102)	Secondary text
Accent	#e67e22	rgb(230,126,34)	Brand colour

I.ii. Dark Mode Colours

Table I.7: Dark mode colour scheme.

Element	Hex Code	RGB	Usage
Background	#333333	rgb(51,51,51)	Page background
Foreground	#fafafa	rgb(250,250,250)	Primary text
Muted	#999999	rgb(153,153,153)	Secondary text
Accent	#e67e22	rgb(230,126,34)	Brand colour

I.iii. Callout Colours (WCAG AA Compliant)

Table I.8: Callout colour accessibility compliance.

Callout Type	Hex Code	Contrast Ratio	Accessibility
Note	#0066cc	7.5:1	AA Large Text, AAA Normal
Tip	#009955	6.8:1	AA Large Text, AAA Normal
Warning	#cc6600	5.2:1	AA Large Text
Important	#cc0000	5.9:1	AA Large Text
Caution	#b38f00	6.1:1	AA Large Text, AAA Normal