ANNEX A · PRACTICAL 7.3 — FROM MARKDOWN TO PDF

Name: _____ **Group:** _____ **Date:** _____

Work in `deb-lab`, inside `~/tramuntana-docs`. Hand in today on Classroom: `practical-7.3-surname.md` and the PDF.

1 Install the two tools

```
sudo apt install -y pandoc curl xz-utils
cd ~
curl -LO https://github.com/typst/typst/releases/latest/download/\
typst-x86_64-unknown-linux-musl.tar.xz
tar -xf typst-x86_64-unknown-linux-musl.tar.xz
mkdir -p ~/.local/bin
cp typst-x86_64-unknown-linux-musl/typst ~/.local/bin/
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.bashrc
source ~/.bashrc
pandoc --version | head -1 ; typst --version
```

The second line of `curl` starts at the left edge. Pandoc: _____ Typst: _____

2 One source, three formats

```
cd ~/tramuntana-docs
pandoc README.md -o /tmp/readme.html
pandoc README.md -o /tmp/readme.docx
pandoc README.md -t typst
```

Open the two files in `/tmp`. On the screen, `# Tramuntana office server` became:

_____

3 The template: the design of the PDF

Create `template.typ` exactly like this, with **your** name. Leave an empty line at the end.

```
// Design of the PDF. The text comes from docs/*.md
#set document(title: "Tramuntana office server")
#set page(paper: "a4", margin: 2.5cm, numbering: "1 / 1")
#set text(lang: "en", size: 11pt)
#set heading(numbering: "1.1")
#show link: underline
#let horizontalrule = line(length: 100%)

#align(center + horizon)[
  #text(size: 24pt, weight: "bold")[Tramuntana office server]

  Technical documentation

  Your Name · 1st DAM · June 2027
]
#pagebreak()
#outline()
#pagebreak()
```

4 The build script

Create `build.sh`:

```
#!/usr/bin/env bash
# Markdown -> Typst -> PDF
set -e
mkdir -p build docs/img
pandoc docs/*.md -t typst -o build/body.typ
cat template.typ build/body.typ > build/tramuntana-docs.typ
cp -r docs/img build/
typst compile build/tramuntana-docs.typ
echo "Built build/tramuntana-docs.pdf"
```

Then:

```
chmod +x build.sh
./build.sh
git status --short
git add template.typ build.sh
git commit -m "Add build script and template"
```

DONE WHEN the PDF in `build/` opens: title page, contents, numbered chapters · `git status --short` did **not** list `build/`.

5 Break it, read the error, repair it

In `template.typ` change `#outline()` to `#outlin()` and run `./build.sh`.

What is wrong, according to Typst? _____ In which line? _____
Repair it without the editor: `git restore template.typ`, then `./build.sh` again.

6 The first real chapter

Create `docs/05-services.md` with the title `# Services`. Document **one** service of your server (for example the Samba share) in five parts: **Purpose** (one sentence: what for, who uses it) · **Installation** (the commands you ran, in a code block) · **Configuration** (the file and the lines you changed, in a code block, and a table: share, path, who can write) · **Check** (a command that proves it works, and the output you expect) · **Sources** (title, publisher, address or `man` page, date). No passwords. Then `./build.sh`, `git add docs/05-services.md`, commit and `git push`.

DONE WHEN the PDF has a chapter Services with a code block and a table, and the forge shows the commit.

REPORT What to hand in — marking out of 10

`practical-7.3-surname.md`: the two versions · the answer of step 2 · the error message of step 5 and one sentence explaining it · the output of `./build.sh` and `git log --oneline | head -5`. Attach the PDF.
Marking: tools 1 · conversions 1 · template and script build the PDF 3 · error explained 1 · chapter 3 · committed and pushed, no `build/` in the repository 1.

EXIT Exit ticket — alone

1. In docs as code, the "real" document is…

- ☐ a) the PDF
- ☐ b) the `.docx`
- ☐ c) the web page
- ☐ d) the Markdown in the repository

2. Pandoc's job in our pipeline is to…

- ☐ a) lay out the pages
- ☐ b) translate Markdown into Typst markup
- ☐ c) upload the PDF
- ☐ d) check the spelling

3. Why is `build/` in `.gitignore`?

- ☐ a) it can be generated again from the sources
- ☐ b) it contains passwords
- ☐ c) Git cannot store PDF files
- ☐ d) it is too old

4. The files in `docs/` start with `01-`, `02-`… because…

- ☐ a) Git needs numbers
- ☐ b) Typst counts them
- ☐ c) `docs/*.md` is read in name order
- ☐ d) it looks tidy