

Uso de Pandoc para la creación de documentación

Irving Montalvo

Ago 14, 2024

Índice

Uso de Pandoc para la creación de documentación	1
Introducción	1
Instalación de Pandoc	2
Lenguaje Markdown	2
Elementos comunes en documentación	2
Notas de pie de página	2
Tablas	2
Figuras	3
Diagramas PlantUML	3
Citar	3
Referencias	4
Documento PDF	4
Extendiendo con LaTeX	5
Fórmula matemática	5
Otros estilos de la página	5
Extendiendo con filtros	5
Aspectos adicionales	5
Conclusión	5
Referencias	6
Anexos	6

Uso de Pandoc para la creación de documentación

Introducción

La creación de documentación en varios ámbitos es de gran importancia, por ello mantener un formato sin añadir complicación a la edición es que se recurre a lenguajes de marcado como Markdown. Pero el estándar de Internet es PDF (del inglés: Portable Document Format) así que se recurre a programas como Pandoc que transforman los archivos de Markdown (.md) a otros formatos (PandocOrg, 2024).

En el presente documento se da una explicación en base a ejemplos del uso de Markdown para obtener documentos con formato y estilo.

Instalación de Pandoc

Pandoc es programado en lenguaje Haskell y se encuentra disponible para las plataformas que permite ejecutar algún compilador de ese lenguaje. Por ello tenemos para Linux, Windows, etc.

```
# Instalar Pandoc en Debian
```

```
sudo apt install -y pandoc
```

```
# --
```

También se requieren otros programas como para completar la transformación de Markdown a PDF:

```
# Instalar paquetes básicos
```

```
sudo apt-get install -y texlive-latex-base texlive-latex-extra
```

```
# Instalar paquetes extras
```

```
sudo apt-get install -y texlive-fonts-recommended texlive-fonts-extra
```

```
# --
```

Lenguaje Markdown

El lenguaje de marcado Markdown creado en 2004, pero con estandarización desde 2012, sirve para dar formato a documentación en plataformas como GitHub. Sin embargo, existen varias extensiones agregado nuevos formatos a elementos que se van utilizando en diferentes contextos, por ejemplo: títulos, notas, bibliografía, etc. Ver Pandoc (2024)

Elementos comunes en documentación

Notas de pie de página

Ejemplo de nota ¹. Nota en la misma línea ²

```
<!-- Nota pie de página -->
```

Ejemplo de nota ^[^1]

^[^1]: Nota de pie de página.

```
<!-- Nota en la misma línea -->
```

Nota en la misma línea ^{^[Nota en la misma línea]}

```
<!------>
```

Tablas

Normal	Derecha	Centro	Izquierda
1 1	1 2	1 3	1 4
2 1	2 2	3 3	2 4
3 1	3 2	3 3	3 4

¹Nota de pie de página.

²Nota en la misma línea

```

<!-- Tabla -->
Normal | Derecha | Centro | Izquierda
--      | --:      | :--:    | :--
1 1     | 1 2      | 1 3     | 1 4
2 1     | 2 2      | 3 3     | 2 4
3 1     | 3 2      | 3 3     | 3 4
<!-->

```

Figuras



Figura 1: Logo de Wikipedia

```

<!-- Agregar una figura -->
![Logo de Wikipedia](./wikipedia.png)
<!-->

```

Diagramas PlantUML



```

<!-- Usando PlantUML en un
bloque de código `plantuml` -->
@startuml
Uno -> Dos: Tres
@enduml
<!-->

```

Citar

Requiere un archivo .bib, ver BibTeX (2024).

```

<!-- Citar -->
@doc2024 o con [@doc2024], ver Anexo 1.
<!-->

```

Referencias

```
<!-- Agregar bibliografía utilizada -->
::: {#refs}
:::
<!-->
```

Documento PDF

Los documentos PDF son de gran importancia por su característica de conservar su estilo sin importar el dispositivo o aplicación donde se visualice. El documento también permite conservar metadatos los cuales podemos configurar:

```
# meta.yml
title: Uso de Pandoc para la creación de documentación
author: Irving Montalvo
date: Ago 14, 2024
papersize: a4
fontsize: 12pt
geometry: margin=2.5cm
keywords: pandoc markdown documentación
subject: Creación de documentación con markdown, fórmulas y gráficos matemáticos

# Activar la tabla de contenidos
toc: true
toc-title: "Índice"

# Archivo de referencias
bibliography: ./references.bib

# Estilo de la bibliografía
csl: ./apa7.csl

# Agregando código LaTeX al inicio
header-includes: |
  \renewcommand{\figurename}{Figura}
# --
```

Para transformar el archivo Markdown a PDF utilizamos:

```
# Transformar Markdown a PDF
pandoc --citeproc --filter pandoc-plantuml \
  --lua-filter ./replace.lua \
  --metadata-file ./meta.yml \
  -o doc.pdf ./README.md ./anexos.md
# --
```

Extendiendo con LaTeX

LaTeX es todo un sistema que permite un mayor detalle en la generación de documentos PDF, pero como se menciona al principio la documentación requiere mantener simplicidad, aquí algunos ejemplos:

Fórmula matemática

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

Donde $ax^2 + bx + c = 0$

```
% Fórmula matemática

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


> Donde  $ax^2 + bx + c = 0$ 
% --
```

Otros estilos de la página

```
% Otros estilos de página
\newpage
% --
```

Extendiendo con filtros

Los filtros permiten agregar nuevas funcionalidades a procesamiento del contenido antes de generar el archivo PDF. Entre los filtros principales tenemos:

```
# Instalar otros filtros en Debian
sudo apt install -y pandoc-citeproc-preamble \
    pandoc-filter-diagram pandoc-plantuml-filter
# --
```

Aspectos adicionales

La documentación de uso académico se rige bajo estándares como APA (2022), Chicago, etc. Estándares que requieren formatos muy específicos y bien documentados.

Conclusión

Buscando la simplicidad en la documentación se usa Markdown y para la presentación formal se hace en un documento PDF, es ahí donde un transformador entre formatos como Pandoc tiene relevancia y en caso de buscar mayor detalle se recurre a sistemas como LaTeX o lenguajes como Haskell y Lua.

Referencias

APA. (2022). *Format documents in APA style (7th edition)*. <https://ctan.org/pkg/apa7>
BibTeX. (2024). *Web official*. <http://www.bibtex.org/Format/>
Pandoc. (2024). *Pandoc user's guide*. <https://pandoc.org/MANUAL.html>
PandocLua. (2024). *Pandoc lua filters*. <https://pandoc.org/lua-filters.html>
PandocOrg. (2024). *Web official*. <https://pandoc.org/>

Anexos

Anexo 1. Archivo de referencias

```
@misc{doc2024,  
  title={Pandoc User's Guide},  
  author={Pandoc},  
  year={2024},  
  url={https://pandoc.org/MANUAL.html}  
}
```

Anexo 2. Filtro con Lua

Filtro que reemplaza `{{ message }}` por “*Oi, Lua!*”, ver PandocLua (2024).

```
-- replace.lua  
return {  
  {  
    Str = function(elem)  
      if elem.text == "{{message}}" then  
        return pandoc.Emph { pandoc.Str "Oi, Lua!" }  
      else  
        return elem  
      end  
    end,  
  },  
}
```