

南臺科技大學

電子工程系碩士班

碩士學位論文

南臺科技大學 L^AT_EX 論文樣板

STUST L^AT_EX Thesis Template

January 31, 2024 (初稿)

研 究 生：您的名字

指導教授：您老師名

中華民國一一三年一月

南臺科技大學

電子工程系碩士班

碩士學位論文

南臺科技大學 L^AT_EX 論文樣板

STUST L^AT_EX Thesis Template

January 31, 2024 (初稿)

您的名字

指導教授： 您老師名

中華民國一一三年一月

摘 要

本專案為南臺科技大學碩士論文 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 樣板，依照南臺科技大學學位論文格示規範，及電子工程系論文格式標準進行編排。

關鍵詞： $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 、樣板、碩士論文



STUST L^AT_EX Thesis Template

Abstract

This project is a template of the master's thesis L^AT_EX of STUST , and it is arranged in accordance with the specifications of STUST's degree thesis format and the format of the electronic engineering department.

Keywords: STUST, Master's Thesis, Template



誌 謝

感謝協助此專案 github.com/tw1chao/STUST-thesis-template 所有貢獻者，
若沒有你們協助可能無法完成此 L^AT_EX 樣板。



目 次

	頁碼
摘 要	i
Abstract	ii
誌 謝	iii
目 次	iv
表 目 錄	vi
圖 目 錄	vii
第一章 緒論	1
1.1 前言	1
1.1.1 為什麼使用 LaTeX?	1
1.2 製作樣板動機	3
1.3 樣板架構	3
第二章 使用此模版並編譯論文	4
2.1 Docker	4
2.1.1 texlive	4
2.2 texlive	4
2.3 MiKTeX	5
2.4 編輯器	5
2.4.1 REMOTE	5
2.4.2 LATEX	5
2.4.3 PDF	6
2.4.4 GIT	6



2.4.5	make	6
2.4.6	cmake	11
第三章	基本排版範例	12
3.1	字體	12
3.1.1	文字特效	12
3.1.2	文字大小	13
3.1.3	文字醒目提示	14
3.1.3.1	文字畫線效果	14
3.1.3.2	盒子	14
3.2	表格	15
3.2.1	小數點對齊	15
3.3	插入圖片	17
3.3.1	插圖範例	17
3.3.2	繪圖範例	18
3.4	公式	18
3.5	利用註腳，收納資料	20
3.6	日文範例	20
3.7	條列式	21
3.8	對齊	22
第四章	演算法	23
4.1	程式碼範例	23
第五章	結果	24
5.1	成功編譯	24
第六章	結論與未來展望	25
6.1	樣板結論	25
6.2	未來期許	25
參考文獻	26



表 目 錄

表 3.1	tabular 表格的基本結構	15
表 3.2	橫著放	16

圖 目 錄

圖 3.1	南臺校園	17
圖 3.2	南臺科技大學	17
圖 3.3	SIN FUNCTION	18



第一章 緒論

1.1 前言

想在畢業之前多學習一些東西，除了常用 office 軟體之外，更想挑戰自我使用 \LaTeX 撰寫畢業論文 [1]。透過 F Mittelbach 所撰寫的 \LaTeX 夥伴一文中 [2] 可以學習到 \LaTeX [3] 的使用方式，其實最大的門檻還是起步的樣板 (模板)，在朝陽科大碩士學位論文研究 [4] 中，使用 Shell 的方式；與本樣板使用 makefile 方式大同小異。

能在畢業前為學校貢獻，並設計南臺科技大學專用碩博士畢業論文模版，公開至 [Github](#) 上，在初學 \LaTeX 時避免語法不熟悉，在本模版中加入大量範例，有效防止學弟妹花費大量時間查找 latex 語法資料。

1.1.1 為什麼使用 \LaTeX ？

使用 Microsoft Office Word 寫論文也可以完成學位呀！為什麼要這麼麻煩？特地使用一個自己不熟悉的軟體進行論文撰寫呢？難到使用 \LaTeX 寫出來的論文就會比較高級嗎？

使用自己熟悉的軟體撰寫文件哪有什麼困難的？你若一直在舒適圈不願意出來看看外面的世界，那你這個即將畢業的學生就有半退休心態的公務員有什麼兩樣呢？據說使用 \LaTeX 在 Linux 上編譯出來的 PDF 檔，是無法逆向回去到 Office Word 格式的，尚未驗證過無法保證，倘若這謠傳是真實的，更能保證論文不被有心人士修改/盜用。

使用 LaTeX 撰寫論文，將好處整理並羅列如下：

- 將內容變成純文字檔，可以更方便透過 git [5] [6] 對論文作控管版本 [7]，在與教授進行論文修改的同時可以將之前所撰寫的紀錄保存起來 [8]。
- 可以將怒氣發洩在註解上 (誤)，能夠透過註解有效紀錄 Todo 事項。
- 與寫程式使用相同的 IDE，能在同一個 IDE 下完成許多事情。
- 不用在意標點符號溢出邊界問題，不用在意交互參照問題，在此樣板中已設定完成。只需專心撰寫論文內容後，在 Terminal 下 Command 就完成 pdf 檔案。
- 整份文件的一致性，讓您的文件看起來更專業。
- 若有想到新的優點再更新上來。

1.2 製作樣板動機

使用 \LaTeX 的過程中，最困難的那一步並不是架設環境，也不是建立文字內容，是文件的版面設定 (邊框多少、浮水印的透明度、套件衝突等...)，為幫助有心使用 \LaTeX 卻沒有時間設定瑣碎事務的人們。也建立自己在畢業前對學校的貢獻度 (或許南台根本不在意?)，也可能畢業十年、廿十年都不會被發現這個放在 github 上設為 public 的 repository。若學校有看到希望能加入到教務處的論文樣板中，讓更多學生能夠看見這份為南臺精心設計的 latex 論文樣板。

1.3 樣板架構

本論文樣板編排方式如下：

- 第1章 說明本製作樣板動機與好處
- 第3章 本樣板精心製作 \LaTeX 語法範例
- 第4章 演算法及程式碼插入語法範例
- 第5章 驗證樣板結果
- 第6章 樣板設計結論與對此樣本的未來期許

第二章 使用此模版並編譯論文

本模版使用 [texlive](#) 編譯，並支援 [Docker](#) 之套件 [texlive/texlive](#) 進行 L^AT_EX 論文編譯。

2.1 Docker

[Docker](#) 是一個開放原始碼平台，用於建置、部署及管理容器化應用程式。瞭解容器、容器與 VM 的比較，以及為何 [Docker](#) 如此被廣泛地採用及使用。

2.1.1 texlive

使用 [texlive/texlive](#) 套件，安裝方式如下

```
1 docker pull texlive/texlive // 下載 texlive 套件
```

2.2 texlive

[Texlive](#) 這邊都是 [Texlive](#) 各國的鏡像下載區，台灣的 latex 鏡像站由元智大學及中正大學所建置，但因為中正大學的鏡像站已經 Not Found 404，元智大學的鏡像站檔案下載的時候會 loss 掉很多檔案，最後我從台灣的鄰近國家下載 [株式會社 KDDI 綜合研究所](#)。

[日本 latex wiki](#)

2.3 MiKTeX

經測試 [MiKTeX](#) 安裝完成能夠使用 `make` 指令編譯本模版，但使用 VSCODE 套件 LaTeX Workshop 編譯時，暫有些問題。

2.4 編輯器

推薦使用 [Visual Studio Code](#) (以下簡稱 VSCODE)，在 VSCODE 中可利用多種套件使工作更便利。

2.4.1 REMOTE

VSCODE 的 [Remote Development](#) 套件包允許您打開容器、遠端電腦或 Windows Linux 子系統 (WSL) 並利用 VSCODE 的完整功能集。由於這使您可以在任何地方建立一個全職的開發環境，您可以：

- 在您部署到的相同操作系統上進行開發，或者使用比本地電腦更大、更快或更專業的硬體。
- 在不同的獨立開發環境之間快速切換並進行更新，而不必擔心影響您的本地機器。
- 通過輕鬆啓動、一致的開發容器幫助新的團隊成員/貢獻者快速獲得生產力。
- 從功能齊全的開發工具中直接從舒適的 Windows 中利用基於 Linux 的工具鏈。

由於遠端開發直接在遠端電腦上運行命令和擴充，因此無需在本地機器上放置源代碼即可獲得這些好處。

2.4.2 LATEX

[LaTeX Workshop](#) 是 Visual Studio Code 套件，旨在為使用 Visual Studio Code 進行 LaTeX 排版提供核心功能。

2.4.3 PDF

[vscode-pdf](#) 為在 VSCODE 直接開啓 PDF，方便 LATEX 編譯完成之 PDF 檔案直接校閱。

2.4.4 GIT

[git](#) 是一個開源的分佈式版本控制系統，用於敏捷高效地處理任何或小或大的項目。

Git 是 Linus Torvalds 為了幫助管理 Linux 核心開發而開發的一個開放源碼的版本控制軟體。

Git 與常用的版本控制工具 CVS, Subversion 等不同，它採用了分佈式版本庫的方式，不必服務器端軟件支持。

幫助論文進行版本控制。

2.4.5 make

欲使用 makefile 編譯此模版，在專案目錄底下新增 makefile 及 main/ 目錄底下新增 makefile。

此檔案為專案根目錄底下 makefile

```
1 # file: ./makefile
2 # Date: 2021/10/15
3 # Name: YingChao
4
5
6 # 主要文件名稱 (不需加副檔名)
7 MAINDOC = main
8
9 # 是否使用預編譯
10 precompiled = 0
11
12 # 編譯文件放置目錄檔案
13 BUILD_DIR = build
14
15 # 設定目標及根目錄
16 OUTPDF = main
17 TARGET = ${BUILD_DIR}/${OUTPDF}.xdv
18
19 # 讀取子目錄
20 SUBDIRS = $(shell ls -l | grep ^d | awk '{if($$9=="$(MAINDOC)")
21             print $$9}')
22
23 # 讀取 pdf 以外的輸出文件
24 OUTFILE=$(shell ls -l $(OUTDIRS) | grep ^- | awk '{if($$9!="$(
25             MAINDOC).pdf") print $$9}')
26
27 # 記住當前根目錄路徑
28 ROOT_DIR=$(shell pwd)
29
30 # commands to compile document
31 MAKE = +make
32 LATEX = xelatex
33 MAKEPDF = xdvipdfmx
34 BIBTEX = biber
35 GHOSTSCRIPT = ghostscript
36 xDVI_PDFMx = -E -o $(OUTPDF).pdf
37
38 # commands parameter
39 FILESNC = -synctex=1
40 FILE_ERR = -file-line-error
41 INTERACT = -interaction=nonstopmode #batchmode
42 HALT_ERR = -halt-on-error
43 NONE_PDF = -no-pdf
44 OUTP_DIR = -output-directory=$(ROOT_DIR)/$(BUILD_DIR)
45 OUTP_DRV = -output-driver='xdvipdfmx-z0' # 此參數設定 PDF 壓縮等級
```



```

45
46 ifeq ($(precompiled), 1)
47 PRECOMPL = -shell-escape "&$(LATEX)"
48
49 TEX_Parameter = $(NONE_PDF) $(OUTP_DIR) $(OUTP_DRV) \
50                $(HALT_ERR) $(PRECOMPL)
51 else
52 TEX_Parameter = $(FILESYNC) $(FILE_ERR) $(INTERACT) \
53                $(HALT_ERR) $(NONE_PDF) $(OUTP_DIR) \
54                $(OUTP_DRV)
55 endif
56
57 # 宣告以下為廣域變數，使子目錄的 makefile 也能讀取以下變數
58 export MAINDOC ROOT_DIR BUILD_DIR OUTP_DIR
59 export LATEX BIBTEX TEX_Parameter precompiled
60 export MAKE MAKEPDF xDVI_PDFMx OUTPDF
61
62 NoPrintDir = --no-print-directory # make 不顯示進入/離開目錄訊息
63
64 all: ${TARGET}
65
66 ${TARGET}:
67     @echo "\033[33m\make\_a\_directory\_\\033[0m"
68     @mkdir -p $(BUILD_DIR)
69     @$(MAKE) -C $(SUBDIRS) $(NoPrintDir)
70
71 clean:
72     @rm -rf $(OUTDIRS)
73
74 staypdf:
75 ifeq ($(strip $(OUTFILE)),)
76     @echo "\033[1;31m\MSG\file\_already\_cleaned\_\\033[0m"
77 else
78     @echo "\033[1;31m\MSG\start\_clean\_file\\033[0m"
79     @cd $(OUTDIRS);\
80     rm $(OUTFILE)
81     @echo "\033[1;32m\MSG\Retain\_pdf\_,\_other\_file\_for\_deleting\_\\033[0m"
82 endif
83
84
85 # https://www.internalpointers.com/post/compress-pdf-file-ghostscript-  
linux
86 # sudo apt install ghostscript
87
88 # -sOwnerPassword=mypassword \
89
90 compress:
91     $(GHOSTSCRIPT) -sDEVICE=pdfwrite \

```



```

92      -dCompatibilityLevel=1.4 \
93      -dPDFSETTINGS=/printer \
94      -dQUIET -dBATCH -dNOPAUSE \
95      -dEmbedAllFonts=false \
96      -dSubsetFonts=true \
97      -dConvertCMYKImagesToRGB=true \
98      -dCompressFonts=true \
99      -sOutputFile=${BUILD_DIR}/${MAINDOC}_out.pdf \
100     ${BUILD_DIR}/${MAINDOC}.pdf
101     @echo -e "\033[32m  [MSG] PDF Size is compressed\033[0m"
102     @cd $(BUILD_DIR);\
103     rm -f *.xdv;\
104     rm -f ${MAINDOC}.pdf;\
105     mv ${MAINDOC}_out.pdf ${MAINDOC}.pdf
106
107
108 .PHONY: clean all

```

此檔案為 main 目錄底下 makefile

```
1  # file: main/makefile
2  # Date: 2022/06/11
3  # Name: YingChao
4
5  # source files
6  TEX_FILES=${wildcard *.tex}
7  XDV_FILES = $(ROOT_DIR)/$(BUILD_DIR)/${MAINDOC}.xdv
8
9  CURRENTDIR=$(shell pwd)
10
11
12  all:
13      @echo "\033[33m$(shell pwd)\033[0m"
14      ifeq ($(precompiled), 1)
15          @$(MAKE) xeINITex # initialize
16          @echo "\033[32m[MSG] \033[0m"
17      endif
18      @$(MAKE) xetex # 1st compiling document
19      @echo "\033[32m[MSG] \033[0m"
20      @$(MAKE) bibtex # reference compiling
21      @echo "\033[32m[MSG] \033[0m"
22      @$(MAKE) xetex # 2nd compiling document
23      @echo "\033[32m[MSG] \033[0m"
24      @$(MAKE) xetex # 3rd compiling document
25      @echo "\033[32m[MSG] \033[0m"
26      @$(MAKE) xDVIPDFMx # Genrated PDF
27      @echo "\033[32m[MSG] \033[0m"
28
29  xetex:
30      $(LATEX) $(TEX_Parameter) $(TEX_FILES)
31
32  bibtex:
33      $(BIBTEX) $(ROOT_DIR)/$(BUILD_DIR)/$(MAINDOC)
34
35  xDVIPDFMx:
36      @cd $(ROOT_DIR)/$(BUILD_DIR);\
37      $(MAKEPDF) -vv $(xDVI_PDFMx) $(XDV_FILES)
38
39  xeINITex:
40      $(LATEX) $(OUTP_DIR) -initialize -jobname="$(MAINDOC)" "&$(LATEX)
41      " mylatexformat.ltx ""$(MAINDOC).tex""
42
43  # https://ctan.math.utah.edu/ctan/tex-archive/macros/latex/contrib/
44  mylatexformat/mylatexformat.pdf
```

透過 **make** 命令執行論文 PDF 的編譯，或是透過 2.4.2 套件的功能編譯本論文模版。

make all 編譯論文 PDF，也可直接使用 **make** 命令。

make clean 清除整個 build 目錄。

make staypdf 清除 build 目錄底下 PDF 以外的檔案。

2.4.6 cmake

動作順序為

mkdir build 建立 build 目錄

cd build 進入 build 目錄

cmake .. cmake 建立編譯專案生成 makefile

make make 命令編譯論文專案

也可以是

cmake -S . -B build

cmake --build build

第三章 基本排版範例

3.1 字體

3.1.1 文字特效

中文文字特效, Hello world!

Hello world!, bold face, 粗體

Hello world!, italic, 斜體

Hello world!, slanted, 傾斜

Hello world!, underline, 底線

英文文字家族, Hello world!

Hello world!, roman, 羅馬

Hello world!, typewriter, 等寬

Hello world!, sans serif

HELLO WORLD!, SMALL CAPS

3.1.2 文字大小

一般文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

文字大小範例, Font Size.

3.1.3 文字醒目提示

黃底黑字。

藍色文字

Red color box and red font.

最後請記得回復成預設黑色字體

3.1.3.1 文字畫線效果

這是底線

這是加倍的底線 (雙底線)

波浪線

~~刪除線~~

.....點點線

3.1.3.2 盒子

Latex 就像 html 一樣，排版小到一個字母，大到一個段落，都可以將其視為一個矩型盒子(box)。

mbox: 一個被組合成盒子的句子

fbox: 一個被組合成有外框盒子的句子

makebox/framebox: 可加上一些設定的盒子

一個被組合成盒子的句子

被組合成有外框盒子的句子還不夠寬

3.2 表格

表 3.1 tabular 表格的基本結構

column1	column2	column3
item1	item2	item3
itemA	itemB	itemC

3.2.1 小數點對齊

	headA	headB	headC	headD
test1	65536	32768	1382.81	998.98
test2	12.457	35.21	321.3	51787.787
test3	211.97	5.257	23.649	744.106

表 3.2 橫著放

	column1	column2	column3	column4	column5	column6	column7	column8
1.	item1	-	-	-	-			
2.	item2	-	-	-			-	-
3.	item3	-	-		-	-		-
4.	item4	-	-	-		-	-	
5.	item5	-	-	-	-			
6.	item6	-	-	-		-	-	-
7.	item7	-	-	-	-		-	* ¹
8.	item8	-	-	-			-	
9.	item9		-					



3.3 插入圖片

L^AT_EX 圖片插入範例區塊。

3.3.1 插圖範例

圖片插入範例：



圖 3.1 南臺校園



(a) 校門



(b) 校徽

圖 3.2 南臺科技大學

圖片插入及圖片並排測試，並排之 subfigure 之間不能有空行。



3.3.2 繪圖範例

繪圖範例 1：

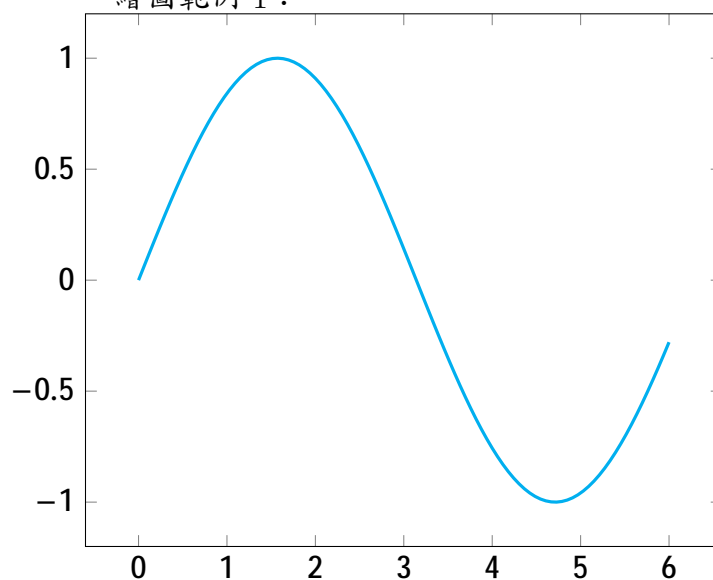


圖 3.3 SIN FUNCTION

正弦波形繪製

3.4 公式

$$\sqrt[n]{\frac{x^2 + 2}{x + y}} \quad (3.1)$$

$$a : b : c = ma : mb : mc = \frac{a}{m} : \frac{b}{m} : \frac{c}{m}, (m \neq 0) \quad (3.2)$$

$$\lim_{n \rightarrow \infty} \sum_{i=1}^n \frac{1}{n} \sin \frac{k}{n} \quad (3.3)$$

$$g(x,y) = \begin{cases} f(x,y), & \text{if } x < y \\ f(y,x), & \text{if } x > y \\ 0, & \text{otherwise.} \end{cases} \quad (3.4)$$

$$\mathbf{A} = \begin{pmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{pmatrix} \quad (3.5)$$

歐姆定律 (Ohm's law)，如式3.6所示， V 是電壓， I 是電流，比例常數 R 是電阻。

$$V = I \cdot R \quad (3.6)$$

3.5 利用註腳，收納資料

腳註，又名註腳、筆記、札記(德語：Fußnote；英語：Footnote)，是讀書所做的筆錄、見解、體會、看法、觀點、心得、感想的總稱。腳註分為：頭注、側注、後注、割注、分注。¹ 在版面的下方，L^AT_EX 會畫一條短線，用比較小的字型將附註的編號和內容安排在短線之下。

3.6 日文範例

設定字體為 google fonts SawarabiMincho (明朝字體)

日本語 (にほんご、にっぽんご) は、主に日本国 や日本人同士の間で使われている言語である。日本は法令によって公用語を規定していないが、法令その他の公用文は全て日本語で記述され、各種法令において日本語を用いることが定められ、学校教育においては「国語」として教えられるなど、事実上、唯一の公用語となっている。

回復預設字體

¹腳註的格式有時候是零散的、即興的、偶然的、信筆成文的，因此它並不是有章法的、系統的、全面的、學術論文式樣的行筆條理清晰。腳註不同於參考文獻(交互參照)，有對於原文若干字句、詞語、典故、暗示、比喻等文筆的疏通和講解。

3.7 條列式

第一大項 這裡是第一大項。

第二大項 這裡是第二大項。

第一小項 這裡是第一小項。

第二小項 這裡是第二小項。

第三大項 這裡是第三大項。

第四大項 這裡是第四大項。

- 這邊是第 1 大項
 - 這邊是第 2 大項
 - 這邊是第 1 小項
 - 這邊是第 2 小項
 - 這邊是第 3 大項
1. 這邊是第 1 大項
 2. 這邊是第 2 大項
 - * 這邊是第 1 小項
 - (a) 這邊是第 2 小項
 - (b) 這邊是第 3 小項
 3. 這邊是第 3 大項

3.8 對齊

本段落

向左對齊

本段落

向右對齊

本段落

置中對齊

第四章 演算法

演算法 1 Heuristic Algorithm (a, b, c).

```
1:  $a =$   
2: while  $U$  is not empty do  
3:   Evaluate  $u$  for all  $u \in U$   
4:   if counter =  $|A|$  then  
5:     break  
6:   end if  
7: end while  
8: return  $a$ ;
```

4.1 程式碼範例

```
1  /* 範例就來個熟悉的 Hello world */  
2  
3  int main(int argc, char ** argv)  
4  {  
5      printf("Hello_world!\n");  
6      return EXIT_SUCCESS;  
7  }
```


第五章 結果

5.1 成功編譯

透過 make 指令能夠協助您有效完成建立 PDF 檔案，建立之 PDF 檔案會在 build 目錄底下。

第六章 結論與未來展望

6.1 樣板結論

期望有後續學弟能夠透過 L^AT_EX 完成畢業論文/專題，順利畢業。
若有其他建議此模板能夠更完善、更美好或是更方便使用，歡迎提交 issues 使專案能夠更人性化。

6.2 未來期許

期望拿到順利拿到學位，通過口試後可以脫離這片土地。
在南台，畢業暨離校手續與退學暨離校手續，同時送出的話，誰會先跑完流程呢？我問過身邊兩三位即將畢業的同學，大家都不願意實驗看看，哪一個流程可以跑得更快。

參考文獻

- [1] 大家來學 *Latex*, <https://jupiter.math.nctu.edu.tw/~smchang/latex/latex123.pdf>, Accessed: 2021-10-07 (page 1).
- [2] F. Mittelbach, M. Goossens, J. Braams, D. Carlisle, and C. Rowley, *The L^A-TEX companion*. Addison-Wesley Professional, 2004 (page 1).
- [3] *The Latex project*, <https://www.latex-project.org/>, Accessed: 2021-09-27 (page 1).
- [4] 陳奕靜, “使用 shell 腳本自動處理 latex 文稿之設計,” 朝陽科技大學資訊工程系學位論文, pp. 1–69, 2012 (page 1).
- [5] S. F. Conservancy. “git –fast-version-control.” (2010), [Online]. Available: <https://git-scm.com/> (visited on 09/27/2021) (page 2).
- [6] 鶴. 彰. (高見龍 (著), やりたいことが今すぐわかる逆引き *Git* 入門. 秀和システム, 2019, ISBN: 4798059595 (page 2).
- [7] D. Spinellis, “Git,” *IEEE software*, vol. 29, no. 3, pp. 100–101, 2012 (page 2).
- [8] S. Chacon and B. Straub, *Pro git*. Springer Nature, 2014 (page 2).