FILES = $(patsubst%.md,%.docx, $(wildcard*.md))$FILES+ =$(patsubst%.md,%.pdf, $(wildcard*.md))

LATEX_FORMAT =

FILTER =–filterpandoc-crossref

%.docx: %.md -pandoc"< "$(FILTER)-o "$@"

%.pdf: %.md -pandoc"< "$(LATEX_FORMAT) $(FILTER) - o"@"

all: $(FILES)@echo$(FILES)

clean: -rm $(FILES)*~

2

David Michael Francis

.   **git.**

1. -   ,   *git   gh,*   *dnf install git   dnf install gh*

Figure 1:   -

2.　　　　　　　　　　　　　　.　　　　　*utf-8*　　　　　　*git-*　　　.
　　　　git-　　.　　　　　*ssh-*

```
Complete!
root@vbox:~# git config --global user.name "Michael David"
root@vbox:~# git config --global user.email "ukanadaushie@gmail.com"
root@vbox:~# git config --global core.quotepath false
root@vbox:~# git config --global init.defaultBranch master
root@vbox:~# git config --global core.autocrlf input
root@vbox:~# git config --global core.safecrlf warn
root@vbox:~# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase for "/root/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:PqD2qYvtkCIW5BI0DaVHiAJ4pbZmr8RB/YVlouyTDIE root@vbox
The key's randomart image is:
+---[RSA 4096]----+
|==*+. . o        |
|E.=* . =         |
|o+=.+ . .        |
|o+.= o .         |
|.o= = o S        |
|.+.+ o o         |
|o.= +   o        |
```

Figure 2: Screenshot 2

3.　　　　　　　*pgp,*　　　　　　　*gpg –full-generate-key.*

4.　　　　　*pgp　　github*

5.　　　　　　　　　　　　　　　*git* git config –global user.signingkey git config
–global commit.gpgsign true git config –global gpg.program $(which gpg2)

6.

**　:**　　　　　　　　　　　　　,　　　　　　git.　　　　　　　　　　.
ssh-　pgn-　.　　　　　　　　　　　　github

　　　　　　　　1.　　　　　　　　　　　(VCS)
　　　?　　　　　　　*(VCS)*　　　　,　　　　　　　　　,
　　　　　　　　　　　.　　　　　　　,　　　　　　,
　　　　.

2.　　　　　VCS　　　:　　, commit,　　,　　　.
　　:　　　,　　　　　.　:　　　　　,
　.　:　　,　　　　.　:　　,
　.

3.　　　　　　　　　　　　VCS?　　　VCS
.

Figure 3: Screenshot 3

Figure 4: SCreenshot 4

```
pub    rsa4096 2025-03-02 [SC]
       0E9F96A906BAF4316C1D8D28E27DFA4588FD67F8
uid                     Michael <omotole47@gmai.com>
sub    rsa4096 2025-03-02 [E]

root@vbox:~# gpg --list-secret-keys --keyid-format LONG
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   1  signed:    0  trust: 0-, 0q, 0n, 0m, 0f, 1u
[keyboxd]
---------
sec    rsa4096/E27DFA4588FD67F8 2025-03-02 [SC]
       0E9F96A906BAF4316C1D8D28E27DFA4588FD67F8
uid                  [ultimate] Michael <omotole47@gmai.com>
ssb    rsa4096/29A1F451E7B997DF 2025-03-02 [E]

root@vbox:~# gpg --armor --export <PGP  Fingerprint> | xclip -sel clip
-bash: syntax error near unexpected token `|'
root@vbox:~# sudo apt install xclip
sudo: apt: command not found
root@vbox:~# gpggg --armor --export E27DFA4588FD67F8
bash: gpggg: command not found...
root@vbox:~# gpg --armor --export E27DFA4588FD67F8
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGfEQggBEADfIc6zw0vEx3DdFXV3QEHWstlJ6qKuMbSTZu5MdeoEgpIpRiAN
```

Figure 5: Screenshot 5

GPG keys

New GPG key

This is a list of GPG keys associated with your account. Remove any keys that you do not recognize.

GPG

Gpg key
Email address: omotole47@gmai.com Unverified
Key ID: E27DFA4588FD67F8
Subkeys: 29A1F451E7B997DF
Added on Mar 2, 2025

Delete

Learn how to generate a GPG key and add it to your account.

Figure 6: SCreenshot 6

Figure 7: Screenshot 7

Figure 8: Screenshot 8

(CVCS): . :
Subverntion (SVN), Perforce. *(DVCS):*
, . *: Git, Mercurial.*

4. VCS

. . .

5. VCS Shared Storage

Clone/pull the latest version. Make changes and commit them locally. Push changes to the shared repository. Resolve conflicts if necessary.

6. , Git

7. Git

Developers: Write and commit code. Maintainers: Review and merge changes. Release Managers: Manage versions and releases.

8. - :

```
git init
git add .
git commit -m "Initial commit"
```

• :

```
git remote add origin <repo_url>
git push origin main
git pull origin main
```

9. : , .

```
git branch feature-branch
git checkout feature-branch
```

10. Git

```
node_modules/
*.log
```

This prevents unnecessary files from being committed.