

Installation and Deployment Instructions

Thomas Kongonis

November 2019

Contents

1	Explanation	1
2	Installation	1
3	Deployment	2
4	Documentation and Review	2

1 Explanation

- This Software deployment doubles at an attempt at creating software and also creating a workflow process with another set of translations.
- Due to this, the software directory contains two separate directories, one is my software deployment coded with my chosen translations. The second named blank workspace is a template set of directories and some simple codes to replicate the same thing with different translations.

2 Installation

1. Setting up workspace:

- Download and extract BlankWorkspace.zip.
- Download and install or open Gitbash and set operating area to the BlankWorkspace directory
- Utilise mv code to move translations into the OriginalTranslations Directory
- If any more than the example amount of translations are present, create extra directories and .txt documents through mkdir and mv functions.

2. Correctly formatting translations and chapters in workspace:

- Utilise Shell Command or gui to implement the respective chapter formation shell scripts which are within each folder. You can either code `bash filename.sh` or double click on it within the gui and it will create the text files enmasse.
- Format chapters pulled from the original translations correctly, including removing any symbols that may cause the script to return errors or not pull the correct files up.

3 Deployment

1. Situate in workspace:

- Open Gitbash and utilise `cd` code to situate in the BlankWorkspace Directory
- Utilise `mv` code to rename the directory to your desired name.

2. Setting up and running Shell Script:

- open script in shell command and situate within shellscripts folder or Translations folder.
- If in Translations folder type command `bash desired chapter code` and press return. For example you would type `bash c05code.sh`
- Run shell script and pull up desired chapters within the shell command.
- gather desired information and repeat as necessary.
- if in shellscript folder, you will need to

4 Documentation and Review

1. Documenting used Shell Scripts

- After changing the shell script, saving it as a new script within the directory will give you the ability to have a script for each chapter that you wish to pull up. This will create less work down the line, especially if old chapters need to be brought up.

2. Documenting failures:

- As this doubles as a workflow, the whole process will also help to enforce very basic shell command skills, therefore any failures should be documented in `failures.txt` document for future reference.