Summer Fellowship Report

On

Development of Windows offline Installer, Ubuntu offline Installer and Update Feature

Submitted by

Adnan Abdullah

Under the guidance of

Prof. Sidhartha Ghosh
Civil Engineering Department
IIT Bombay

Under the Mentorship of

Danish Ansari
Project Research Associate

Aug 22, 2021

Acknowledgment

I would like to thank FOSSEE for providing me a platform to work on something I am very interested in. I am thankful to everyone who thought of having and involved in selection process based on screening tasks. I am grateful to be a part of team which promotes open source software. I thank all the Osdag members, who are wonderful mentors and great team. I thank Danish Ansari (Project Research Assistant) who guided me at every point.

Contents

Chapter 1

Introduction

1.1 Osdag Internship

Osdag internship is provided under the FOSSEE project. FOSSEE project promotes the use of FOSS (Free/Libreand Open Source Software) tools to improve quality of education in our country. FOSSEE encourages the use of FOSS tools through various activities to ensure availability of competent free software equivalent to commercial (paid) softwares.

The FOSSEE project is a part of the National Mission on Education through Infrastructure and Communication Technology(ICT), Ministry of Human Resources and Development, Government of India.

Osdag is one such open source software which comes under the FOSSEE project. Osdag internship is provided through FOSSEE project. Any UG/PG/PhD holder can apply forthis internship. And the selection will be based on a screen- ing task.

Osdag
Open steel design and graphics

1.2 What is Osdag?

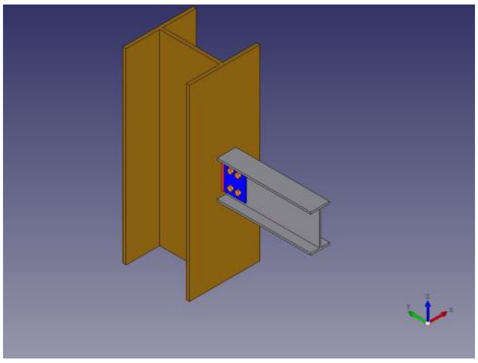
Osdag is Free/Libre and Open Source Software being developed for design of steel structures. Its source code is written in Python, 3D CAD images are developed using PythonOCC. Github is used to ensure smooth workflow between different modules and team members. It is in a path where people from around the world would be able to contribute to its development. FOSSEE's "Share alike" policy would improve the standard of the software when the source code is further modified based on the industrial and educational needs across the country.

Design and Detailing Checklist (DDCL) for different connections, members and structure designs is one of the important bi-products of this project. It would create a repository and design guide book for steel construction based on Indian Standard codes and best industry practices.

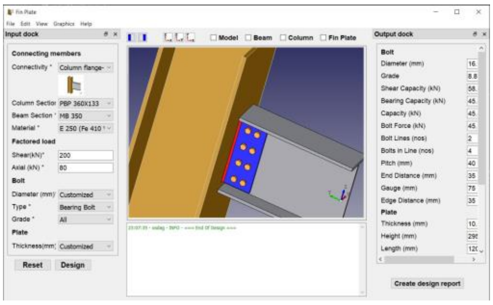
Osdag provides various features such as:

An interactive window displaying a 3D CAD model, which provides a clear visualisation of the designed component.

Creation of 3D CAD models that can be imported to generic CAD softwares.



User-friendly input and output docs, with text-validated fields grouped according to the design flow.


Fin Plate
File Edit View Graphics Help
□ Model □ Beam □ Column □ Fin Plate
Reset
Design
Create design report

Input dock

Connecting members

Connectivity * Column flange-

Column Sectior PBP 360X133

Beam Section * MB 350

Material * E 250 (Fe 410 '

Factored load

Shear(kN)* 200

Axial (kN) * 80

Bolt

Diameter (mm) Customized

Type * Bearing Bolt

Grade * All

Plate

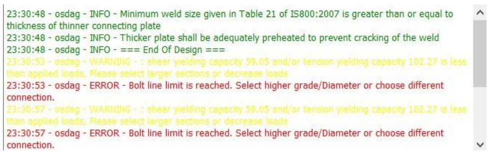
Thickness(mm) Customized

```
23:07:35 - osdag - INFO - === End Of Design ===
```

Output dock

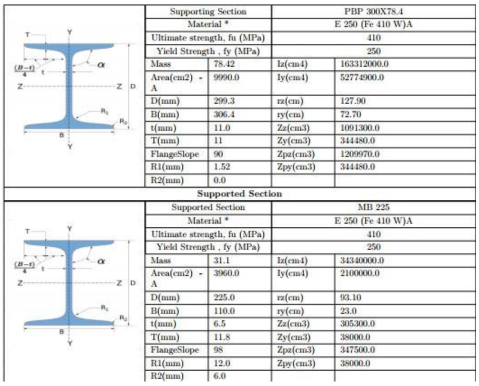
Bolt	
Diameter (mm)	16.
Grade	8.8
Shear Capacity (kN)	58.
Bearing Capacity (kN)	45.
Capacity (kN)	45.
Bolt Force (kN)	45
Bolt Lines (nos)	2
Bolts in Line (nos)	4
Pitch (mm)	40
End Distance (mm)	35
Gauge (mm)	75
Edge Distance (mm)	35
Plate	
Thickness (mm)	10.
Height (mm)	29(
Length (mm)	12(

A text window for message display, that also suggests necessary changes if a trial design is found unsafe.



```
23:30:48 - osdag - INFO - Minimum weld size given in Table 21 of IS800:2007 is greater than or equal to
thickness of thinner connecting plate
23:30:48 - osdag - INFO - Thicker plate shall be adequately preheated to prevent cracking of the weld
23:30:48 - osdag - INFO - === End Of Design ===
23:30:53 - osdag - WARNING - : shear yielding capacity 59.05 and/or tension yielding capacity 102.27 is less
than applied loads, Please select larger sections or decrease loads
23:30:53 - osdag - ERROR - Bolt line limit is reached. Select higher grade/Diameter or choose different
connection.
23:30:57 - osdag - WARNING - : shear yielding capacity 59.05 and/or tension yielding capacity 102.27 is less
than applied loads, Please select larger sections or decrease loads
23:30:57 - osdag - ERROR - Bolt line limit is reached. Select higher grade/Diameter or choose different
connection.
```

Creation of a professional design report showing all necessary checks, design calculations as per IS 800:2007, and standard views of the designed component.


T
Y
$\frac{(B-t)}{4}$
t
α
Z
Z
D
R1
R2
B
Y

Supporting Section		PBP 300X78.4	
Material *		E 250 (Fe 410 W)A	
Ultimate strength, fu (MPa)		410	
Yield Strength , fy (MPa)		250	
Mass	78.42	Iz(cm4)	163312000.0
Area(cm2) - A	9990.0	Iy(cm4)	52774900.0
D(mm)	299.3	rz(cm)	127.90
B(mm)	306.4	ry(cm)	72.70
t(mm)	11.0	Zz(cm3)	1091300.0
T(mm)	11	Zy(cm3)	344480.0
FlangeSlope	90	Zpz(cm3)	1209970.0
R1(mm)	1.52	Zpy(cm3)	344480.0
R2(mm)	0.0		

Supported Section

Supported Section		MB 225	
Material *		E 250 (Fe 410 W)A	
Ultimate strength, fu (MPa)		410	
Yield Strength , fy (MPa)		250	
Mass	31.1	Iz(cm4)	34340000.0
Area(cm2) - A	3960.0	Iy(cm4)	2100000.0
D(mm)	225.0	rz(cm)	93.10
B(mm)	110.0	ry(cm)	23.0
t(mm)	6.5	Zz(cm3)	305300.0
T(mm)	11.8	Zy(cm3)	38000.0
FlangeSlope	98	Zpz(cm3)	347500.0
R1(mm)	12.0	Zpy(cm3)	38000.0
R2(mm)	6.0		

1.3 Who can use ?

Osdag is created both for educational purpose and industry professionals. As Osdag is currently funded by MHRD,

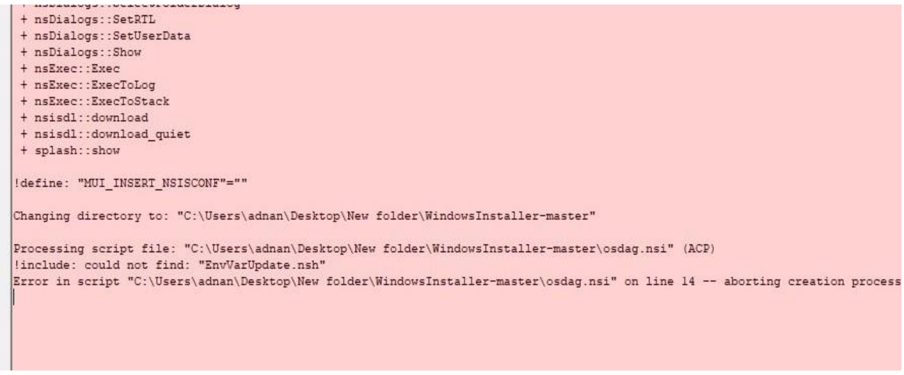
Osdag team is developing software in such a way that it can be used by the students during their academics and to give them a better insight look in the subject.
Osdag can be used by anyone starting from novice to professionals. It’s simple user interface makes it flexible and attractive than other software. Video tutorials are available to help get started. The video tutorials of Osdag can be accessed here.

Chapter 2

Development work on Windows Offline Installer

2.1 Removed error and warning from from the nsis script

When I start compiling nsis script I encountered with the first error that was !include: could not find : “EnvVarUpdate.nsh” and !include: could not find: “RefreshEnvironment.nsh”.

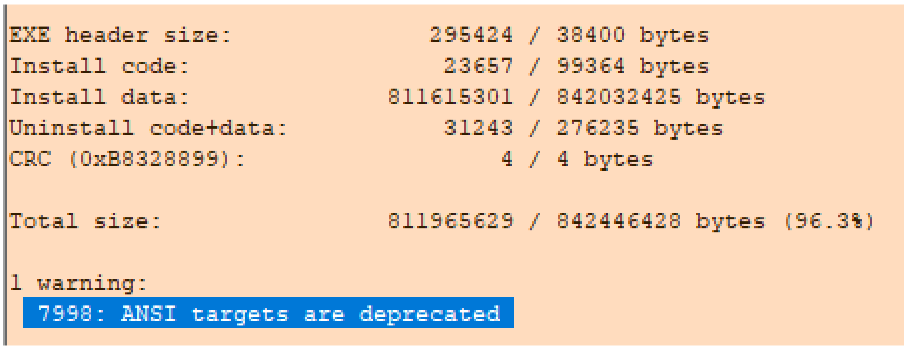


```
 + nsDialogs::SelectFolderDialog
 + nsDialogs::SetRTL
 + nsDialogs::SetUserData
 + nsDialogs::Show
 + nsExec::Exec
 + nsExec::ExecToLog
 + nsExec::ExecToStack
 + nsisdl::download
 + nsisdl::download_quiet
 + splash::show

!define: "MUI_INSERT_NSISCONF"=""

Changing directory to: "C:\Users\adnan\Desktop\New folder\WindowsInstaller-master"

Processing script file: "C:\Users\adnan\Desktop\New folder\WindowsInstaller-master\osdag.nsi" (ACP)
!include: could not find: "EnvVarUpdate.nsh"
Error in script "C:\Users\adnan\Desktop\New folder\WindowsInstaller-master\osdag.nsi" on line 14 -- aborting creation process
```

Then I fixed these error by writing

```
!include header_files\EnvVarUpdate.nsh
!include header_files\RefreshEnvironment.nsh
```

Now the second I got was the warning in the script “ANSI targets are deprecated”

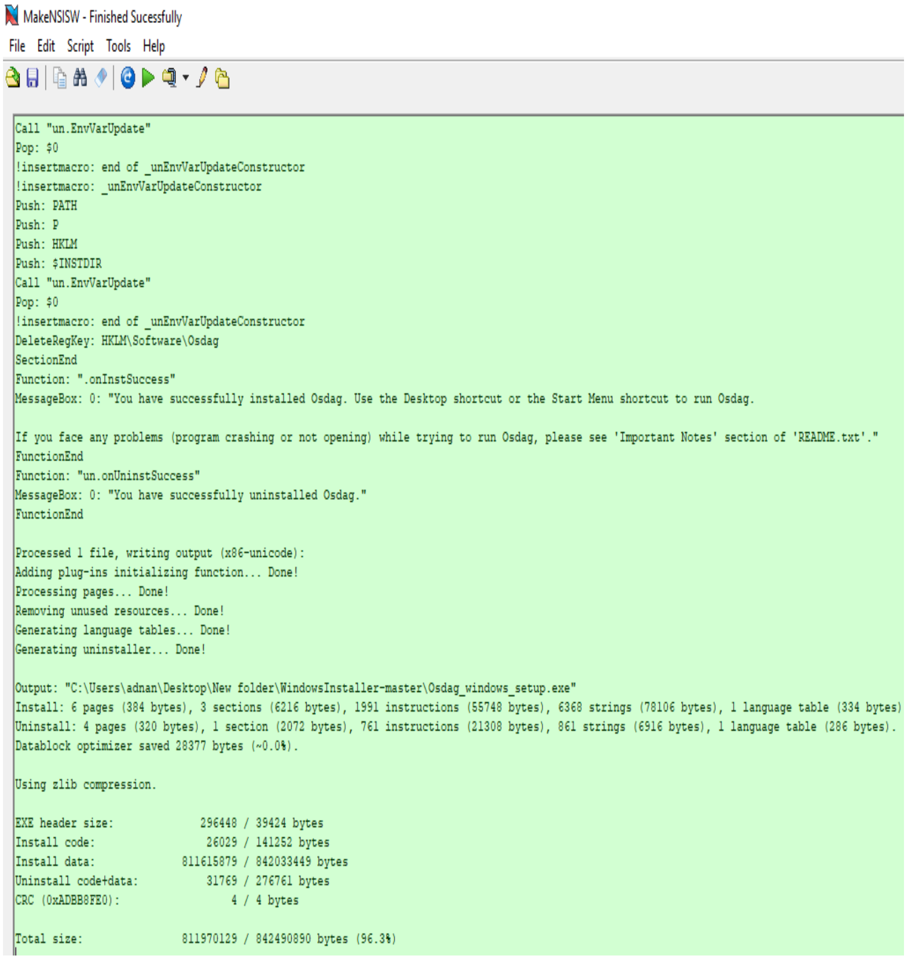


```
EXE header size:              295424 / 38400 bytes
Install code:                  23657 / 99364 bytes
Install data:              811615301 / 842032425 bytes
Uninstall code+data:           31243 / 276235 bytes
CRC (0xB8328899):                  4 / 4 bytes

Total size:                811965629 / 842446428 bytes (96.3%)

1 warning:
  7998: ANSI targets are deprecated
```

and I fixed that warning by adding code line in the nsis script

Unicode true

Now the nsis script was getting compiled and executable was getting generated smoothly



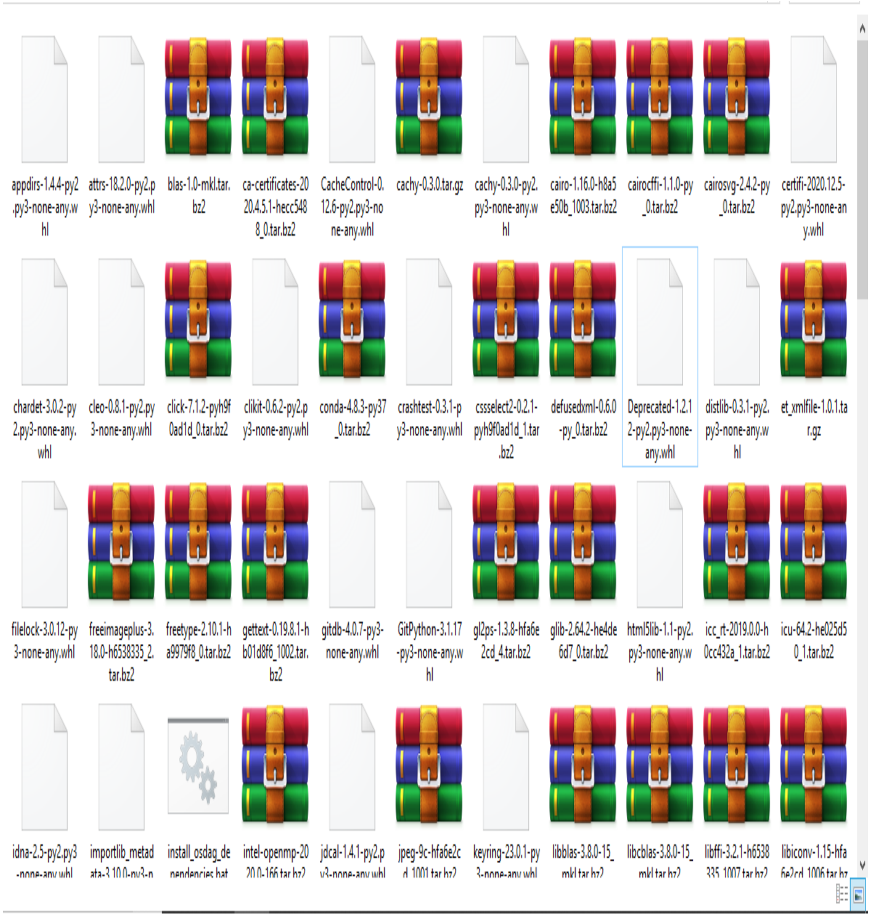
```
Call "un.EnvVarUpdate"
Pop: $0
!insertmacro: end of _unEnvVarUpdateConstructor
!insertmacro: _unEnvVarUpdateConstructor
Push: PATH
Push: P
Push: HKLM
Push: $INSTDIR
Call "un.EnvVarUpdate"
Pop: $0
!insertmacro: end of _unEnvVarUpdateConstructor
DeleteRegKey: HKLM\Software\Osdag
SectionEnd
Function: ".onInstSuccess"
MessageBox: 0: "You have successfully installed Osdag. Use the Desktop shortcut or the Start Menu shortcut to run Osdag.

If you face any problems (program crashing or not opening) while trying to run Osdag, please see 'Important Notes' section of 'README.txt'."
FunctionEnd
Function: "un.onUninstSuccess"
MessageBox: 0: "You have successfully uninstalled Osdag."
FunctionEnd

Processed 1 file, writing output (x86-unicode):
Adding plug-ins initializing function... Done!
Processing pages... Done!
Removing unused resources... Done!
Generating language tables... Done!
Generating uninstaller... Done!

Output: "C:\Users\adnan\Desktop\New folder\WindowsInstaller-master\Osdag_windows_setup.exe"
Install: 6 pages (384 bytes), 3 sections (6216 bytes), 1991 instructions (55748 bytes), 6368 strings (78106 bytes), 1 language table (334 bytes)
Uninstall: 4 pages (320 bytes), 1 section (2072 bytes), 761 instructions (21308 bytes), 861 strings (6916 bytes), 1 language table (286 bytes).
Datablock optimizer saved 28377 bytes (~0.0%).

Using zlib compression.

EXE header size:              296448 / 39424 bytes
Install code:                  26029 / 141252 bytes
Install data:              811615879 / 842033449 bytes
Uninstall code+data:           31769 / 276761 bytes
CRC (0xADBB8FE0):                  4 / 4 bytes

Total size:                811970129 / 842490890 bytes (96.3%)
```

2.2 Added new dependencies

So when tried to install the osdag I encountered with errors That some dependencies are out dated and some are missing And few errors of them I have shown below

- ERROR: No matching distribution found for poetry>=0.12
- ERROR: pandas-1.0.5-cp37-cp37m-win_amd64.whl is not a supported wheel on this platform.
- ERROR: No matching distribution found for setuptools>=42
- ERROR: Command errored out with exit status 1
- ERROR: Could not find a version that satisfies the requirement ordered-set (from pylatex)
- ERROR: No matching distribution found for ordered-set

To fix these errors I collected required dependencies and added them to dependencies folder and screen shots are added below



appdirs-1.4.4-py2.py3-none-any.whl
attrs-18.2.0-py2.py3-none-any.whl
blas-1.0-mkl.tar.bz2
ca-certificates-2020.4.5.1-hecc5488_0.tar.bz2
CacheControl-0.12.6-py2.py3-none-any.whl
cachy-0.3.0.tar.gz
cachy-0.3.0-py2.py3-none-any.whl
cairo-1.16.0-h8a5e50b_1003.tar.bz2
cairocffi-1.1.0-py_0.tar.bz2
cairosvg-2.4.2-py_0.tar.bz2
certifi-2020.12.5-py2.py3-none-any.whl

chardet-3.0.2-py2.py3-none-any.whl
cleo-0.8.1-py2.py3-none-any.whl
click-7.1.2-pyh9f0ad1d_0.tar.bz2
clikit-0.6.2-py2.py3-none-any.whl
conda-4.8.3-py37_0.tar.bz2
crashtest-0.3.1-py3-none-any.whl
cssselect2-0.2.1-pyh9f0ad1d_1.tar.bz2
defusedxml-0.6.0-py_0.tar.bz2
Deprecated-1.2.12-py2.py3-none-any.whl
distlib-0.3.1-py2.py3-none-any.whl
et_xmlfile-1.0.1.tar.gz

filelock-3.0.12-py3-none-any.whl
freeimageplus-3.18.0-h6538335_2.tar.bz2
freetype-2.10.1-ha9979f8_0.tar.bz2
gettext-0.19.8.1-hb01d8f6_1002.tar.bz2
gitdb-4.0.7-py3-none-any.whl
GitPython-3.1.17-py3-none-any.whl
gl2ps-1.3.8-hfa6e2cd_4.tar.bz2
glib-2.64.2-he4de6d7_0.tar.bz2
html5lib-1.1-py2.py3-none-any.whl
icc_rt-2019.0.0-h0cc432a_1.tar.bz2
icu-64.2-he025d50_1.tar.bz2

idna-2.5-py2.py3-none-any.whl
importlib_metadata-3.10.0-py3-n
install_osdag_dependencies.bat
intel-openmp-2020.0-166.tar.bz2
jdcal-1.4.1-py2.py3-none-any.whl
jpeg-9c-hfa6e2cd_1001.tar.bz2
keyring-23.0.1-py3-none-any.whl
libblas-3.8.0-15_mkl.tar.bz2
libcblas-3.8.0-15_mkl.tar.bz2
libffi-3.2.1-h6538335_1007.tar.bz2
libiconv-1.15-hfa6e2cd_1006.tar.bz

3.3 fixed error in windows installation script

After adding dependencies succesfully I faced another error “path not found error” while installing those dependencies which was coming from \WindowsInstallermaster\Files\dependencies\install_osdag_dependencies.bat

Then I modified the install_osdag_dependencies.bat as below

```
@echo off
set current_path=%~dp0
call pip install %current_path%pynput-1.7.3-py2.py3-none-any.whl
call pip install %current_path%wrapt-1.12.1.tar.gz
call pip install %current_path%python_dateutil-2.8.1-py2.py3-none-any.whl
```

%~dp0 fetches the current directory path from where the dependencies are getting installed.

4.4 fixed error with the database

Now the next error I had to tackle was the error “UNABLE TO OPEN DATABASE” in gusset_connection.py And this database is located at \Files\Osdag\ResourceFiles\Database , so in order to fix this error I did the modification in gusset_connection.py . Lines that I added/modified are given below

```
import pathlib
file_path = os.path.join(os.getcwd(), 'folder', 'filename')
file_path=str(file_path)+\\Intg_osdag.sqlite
```

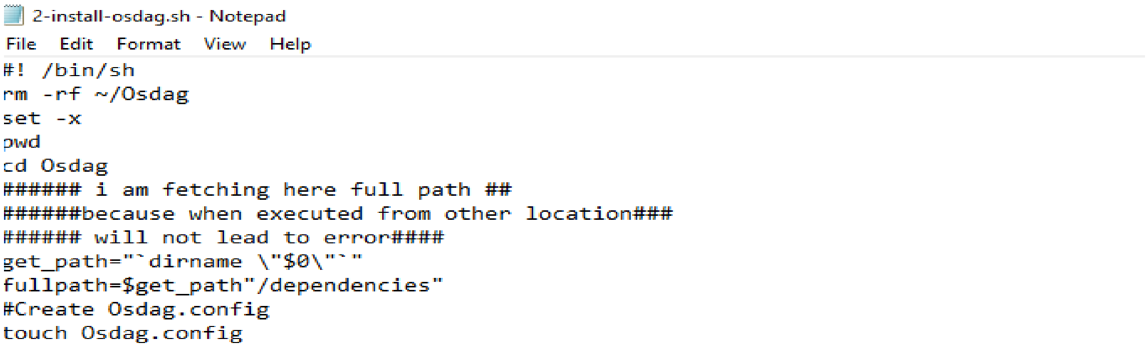
After eliminating the above mentioned errors Osdag started launching smoothly.

Chapter 3

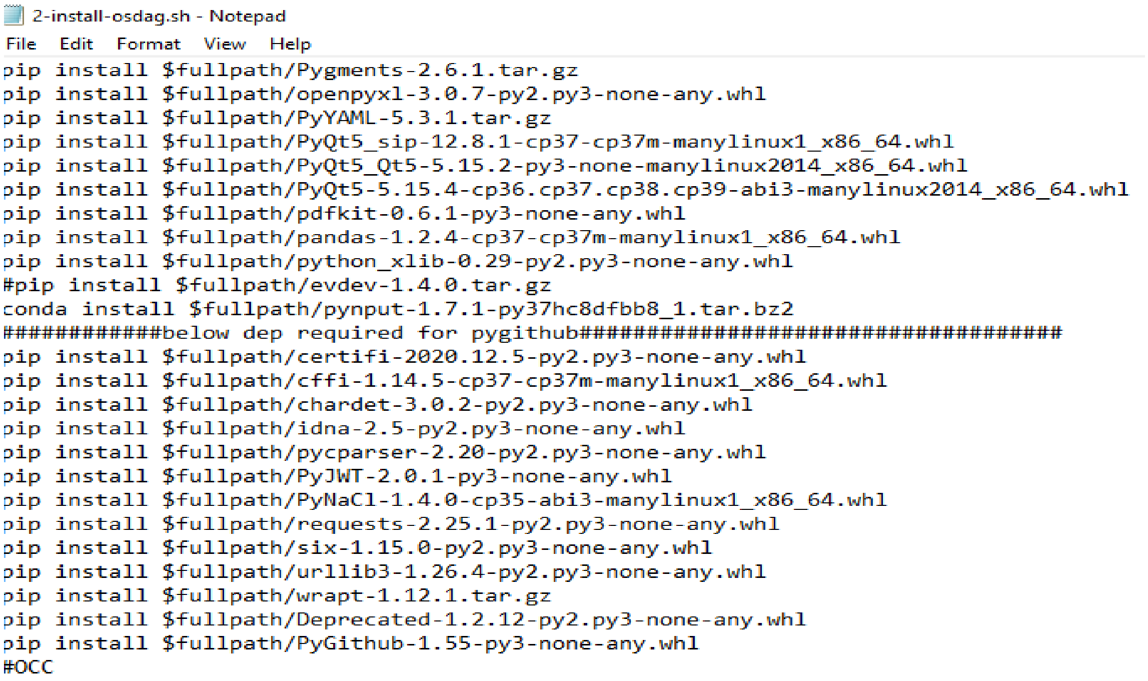
Development work on Ubuntu offline installer

3.1 Writing the installation Script

The previous installation script was not working according to to expectations like it was not generating Osdag icon etc. So I decided to rewrite it again. Which is namely 2-install-osdag.sh . Below is the screenshot of installation script



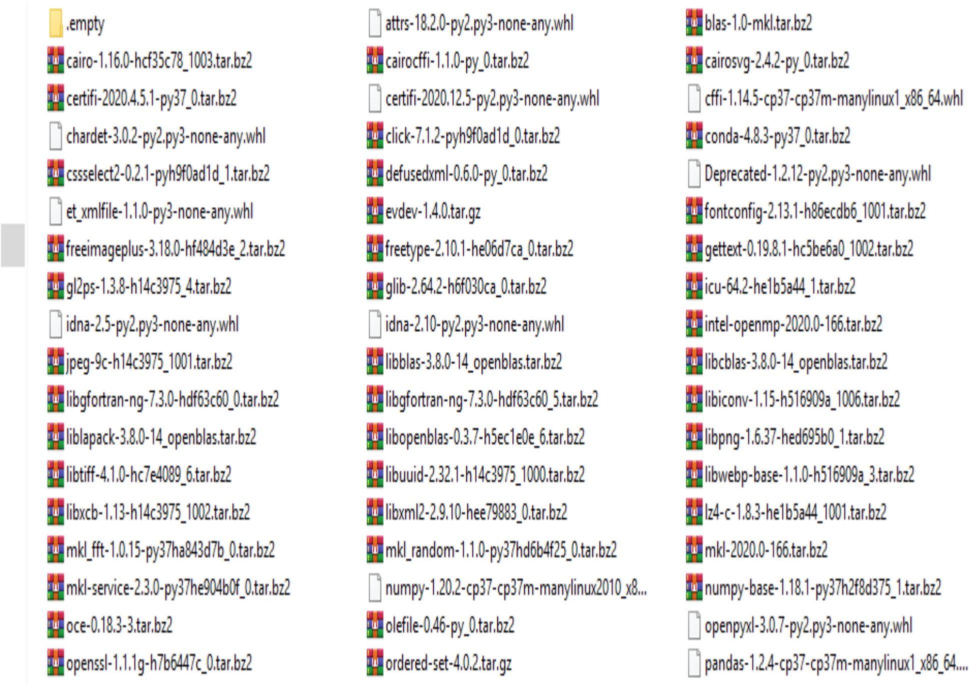
```
#! /bin/sh
rm -rf ~/Osdag
set -x
pwd
cd Osdag
###### i am fetching here full path ##
######because when executed from other location###
###### will not lead to error####
get_path="`dirname \"$0\"`"
fullpath=$get_path"/dependencies"
#Create Osdag.config
touch Osdag.config
```



```
pip install $fullpath/Pygments-2.6.1.tar.gz
pip install $fullpath/openpyxl-3.0.7-py2.py3-none-any.whl
pip install $fullpath/PyYAML-5.3.1.tar.gz
pip install $fullpath/PyQt5_sip-12.8.1-cp37-cp37m-manylinux1_x86_64.whl
pip install $fullpath/PyQt5_Qt5-5.15.2-py3-none-manylinux2014_x86_64.whl
pip install $fullpath/PyQt5-5.15.4-cp36.cp37.cp38.cp39-abi3-manylinux2014_x86_64.whl
pip install $fullpath/pdfkit-0.6.1-py3-none-any.whl
pip install $fullpath/pandas-1.2.4-cp37-cp37m-manylinux1_x86_64.whl
pip install $fullpath/python_xlib-0.29-py2.py3-none-any.whl
#pip install $fullpath/evdev-1.4.0.tar.gz
conda install $fullpath/pynput-1.7.1-py37hc8dfbb8_1.tar.bz2
############below dep required for pygithub######################################
pip install $fullpath/certifi-2020.12.5-py2.py3-none-any.whl
pip install $fullpath/cffi-1.14.5-cp37-cp37m-manylinux1_x86_64.whl
pip install $fullpath/chardet-3.0.2-py2.py3-none-any.whl
pip install $fullpath/idna-2.5-py2.py3-none-any.whl
pip install $fullpath/pycparser-2.20-py2.py3-none-any.whl
pip install $fullpath/PyJWT-2.0.1-py3-none-any.whl
pip install $fullpath/PyNaCl-1.4.0-cp35-abi3-manylinux1_x86_64.whl
pip install $fullpath/requests-2.25.1-py2.py3-none-any.whl
pip install $fullpath/six-1.15.0-py2.py3-none-any.whl
pip install $fullpath/urllib3-1.26.4-py2.py3-none-any.whl
pip install $fullpath/wrapt-1.12.1.tar.gz
pip install $fullpath/Deprecated-1.2.12-py2.py3-none-any.whl
pip install $fullpath/PyGithub-1.55-py3-none-any.whl
#OCC
```

3.2 Added new dependencies

Like I added dependencies in windows, similarly I collected added dependencies for linux installation.



```
.empty
cairo-1.16.0-hcf35c78_1003.tar.bz2
certifi-2020.4.5.1-py37_0.tar.bz2
chardet-3.0.2-py2.py3-none-any.whl
cssselect2-0.2.1-pyh9f0ad1d_1.tar.bz2
et_xmlfile-1.1.0-py3-none-any.whl
freeimageplus-3.18.0-hf484d3e_2.tar.bz2
gl2ps-1.3.8-h14c3975_4.tar.bz2
idna-2.5-py2.py3-none-any.whl
jpeg-9c-h14c3975_1001.tar.bz2
libgfortran-ng-7.3.0-hdf63c60_0.tar.bz2
liblapack-3.8.0-14_openblas.tar.bz2
libtiff-4.1.0-hc7e4089_6.tar.bz2
libxcb-1.13-h14c3975_1002.tar.bz2
mkl_fft-1.0.15-py37ha843d7b_0.tar.bz2
mkl-service-2.3.0-py37he904b0f_0.tar.bz2
oce-0.18.3-3.tar.bz2
openssl-1.1.1g-h7b6447c_0.tar.bz2
attrs-18.2.0-py2.py3-none-any.whl
cairocffi-1.1.0-py_0.tar.bz2
certifi-2020.12.5-py2.py3-none-any.whl
click-7.1.2-pyh9f0ad1d_0.tar.bz2
defusedxml-0.6.0-py_0.tar.bz2
evdev-1.4.0.tar.gz
freetype-2.10.1-he06d7ca_0.tar.bz2
glib-2.64.2-h6f030ca_0.tar.bz2
idna-2.10-py2.py3-none-any.whl
libblas-3.8.0-14_openblas.tar.bz2
libgfortran-ng-7.3.0-hdf63c60_5.tar.bz2
libopenblas-0.3.7-h5ec1e0e_6.tar.bz2
libuuid-2.32.1-h14c3975_1000.tar.bz2
libxml2-2.9.10-hee79883_0.tar.bz2
mkl_random-1.1.0-py37hd6b4f25_0.tar.bz2
numpy-1.20.2-cp37-cp37m-manylinux2010_x8...
olefile-0.46-py_0.tar.bz2
ordered-set-4.0.2.tar.gz
blas-1.0-mkl.tar.bz2
cairosvg-2.4.2-py_0.tar.bz2
cffi-1.14.5-cp37-cp37m-manylinux1_x86_64.whl
conda-4.8.3-py37_0.tar.bz2
Deprecated-1.2.12-py2.py3-none-any.whl
fontconfig-2.13.1-h86ecdb6_1001.tar.bz2
gettext-0.19.8.1-hc5be6a0_1002.tar.bz2
icu-64.2-he1b5a44_1.tar.bz2
intel-openmp-2020.0-166.tar.bz2
libcblas-3.8.0-14_openblas.tar.bz2
libiconv-1.15-h516909a_1006.tar.bz2
libpng-1.6.37-hed695b0_1.tar.bz2
libwebp-base-1.1.0-h516909a_3.tar.bz2
lz4-c-1.8.3-he1b5a44_1001.tar.bz2
mkl-2020.0-166.tar.bz2
numpy-base-1.18.1-py37h2f8d375_1.tar.bz2
openpyxl-3.0.7-py2.py3-none-any.whl
pandas-1.2.4-cp37-cp37m-manylinux1_x86_64....
```

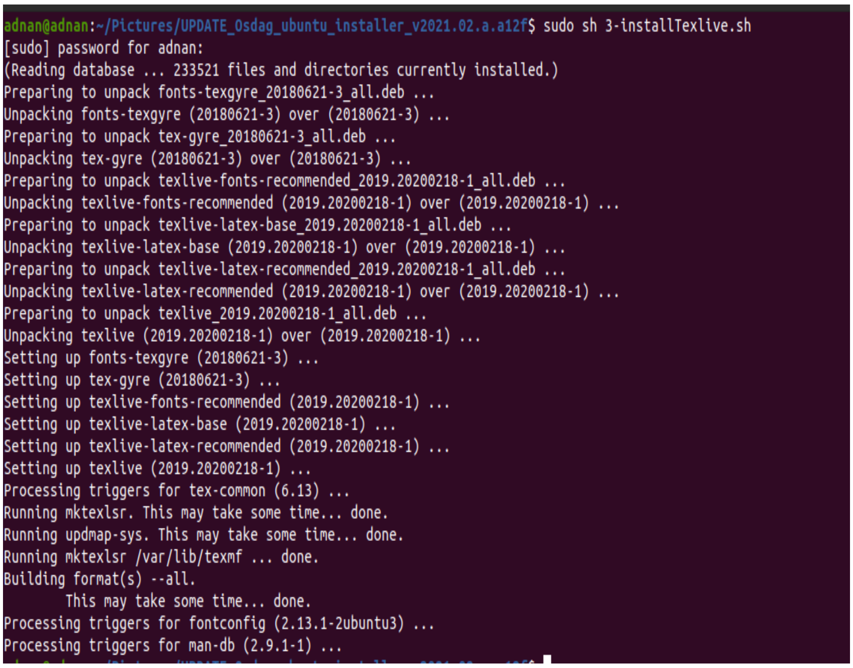
Adding required dependencies made error free installation of Osdag possible.

3.2 Installing TEX Live locally

TeX Live is intended to be a straightforward way to get up and running with the TeX document production system. It provides a comprehensive TeX system with binaries for linux. Installing this locally (without internet) was my biggest challenge Because it offline package is around 2 GB which is alone twice the the size of Osdag. So I wrote a script to do make this job done and also made sure that Osdag size should not grow exponentially. Below is the above said script.

```
get_path="`dirname \"$0\"`"
fullpath=$get_path"/texlivePackages"
cd $fullpath
sudo cp * /var/cache/apt/archives
sudo dpkg -i *
cd ..
```

The best thing about this script is that, from 2GBs tex live size is reduced to 36.3 MB. Below is the screen shot of installation of tex live.



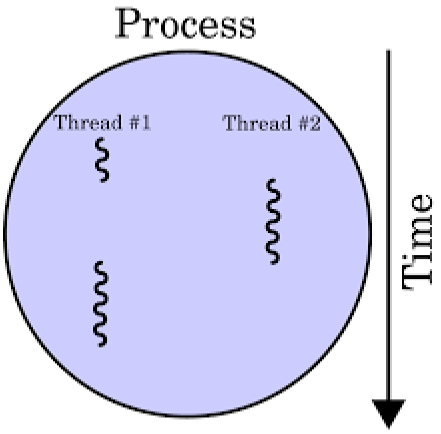
```
adnan@adnan:~/Pictures/UPDATE_Osdag_ubuntu_installer_v2021.02.a.a12f$ sudo sh 3-installTexlive.sh
[sudo] password for adnan:
(Reading database ... 233521 files and directories currently installed.)
Preparing to unpack fonts-texgyre_20180621-3_all.deb ...
Unpacking fonts-texgyre (20180621-3) over (20180621-3) ...
Preparing to unpack tex-gyre_20180621-3_all.deb ...
Unpacking tex-gyre (20180621-3) over (20180621-3) ...
Preparing to unpack texlive-fonts-recommended_2019.20200218-1_all.deb ...
Unpacking texlive-fonts-recommended (2019.20200218-1) over (2019.20200218-1) ...
Preparing to unpack texlive-latex-base_2019.20200218-1_all.deb ...
Unpacking texlive-latex-base (2019.20200218-1) over (2019.20200218-1) ...
Preparing to unpack texlive-latex-recommended_2019.20200218-1_all.deb ...
Unpacking texlive-latex-recommended (2019.20200218-1) over (2019.20200218-1) ...
Preparing to unpack texlive_2019.20200218-1_all.deb ...
Unpacking texlive (2019.20200218-1) over (2019.20200218-1) ...
Setting up fonts-texgyre (20180621-3) ...
Setting up tex-gyre (20180621-3) ...
Setting up texlive-fonts-recommended (2019.20200218-1) ...
Setting up texlive-latex-base (2019.20200218-1) ...
Setting up texlive-latex-recommended (2019.20200218-1) ...
Setting up texlive (2019.20200218-1) ...
Processing triggers for tex-common (6.13) ...
Running mktexlsr. This may take some time... done.
Running updmap-sys. This may take some time... done.
Running mktexlsr /var/lib/texmf ... done.
Building format(s) --all.
        This may take some time... done.
Processing triggers for fontconfig (2.13.1-2ubuntu3) ...
Processing triggers for man-db (2.9.1-1) ...
```

Chapter 4

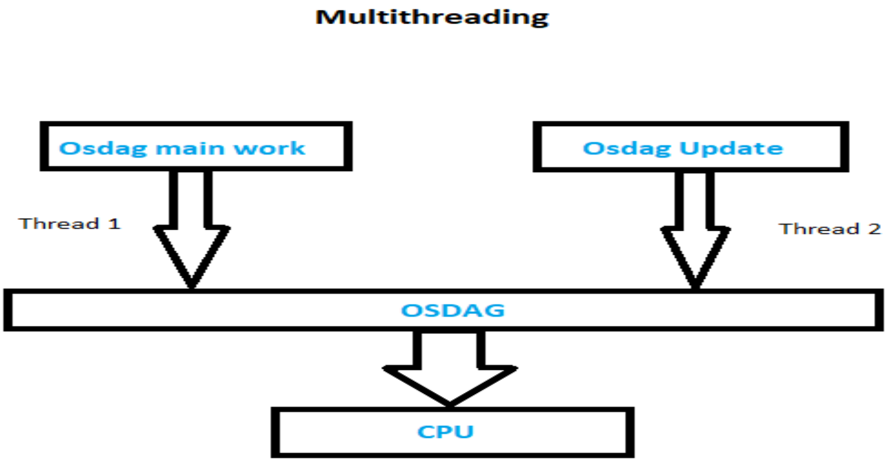
Development work on Update feature

4 . 1 Multi threading and its need in Osdag

Multithreading is a model of program execution that allows for multiple threads to be created within a process, executing independently but concurrently sharing process resources.



Multi threading was needed in Osdag in order to implement Update feature because without multi threading , when ever I tried to update the Gui was freezing.



4 . 2 Check for update and install the update

I have implemented update function in two parts

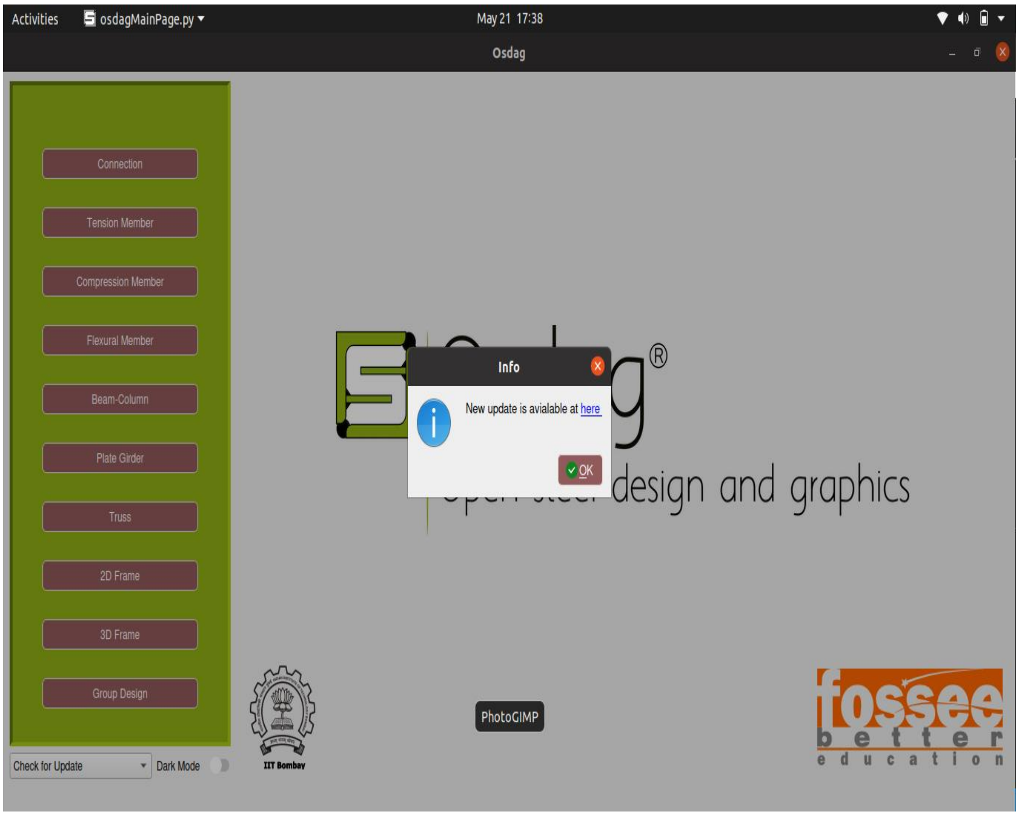
- Major update
- Minor update

Major update is usually released after two years and this is uploaded to Osdag website. And Mini update is frequently released and its code is uploaded to github.com.

so I implemented these two types of updates with following logic

```
if majorUpdate exists
     major update exists and go to osdag website
    and download manually via web browser
else if noOfCommitsOnGihub > noOfcommitsLocally
     mini update exists and click to download
    and install update
else
    no update exists
```

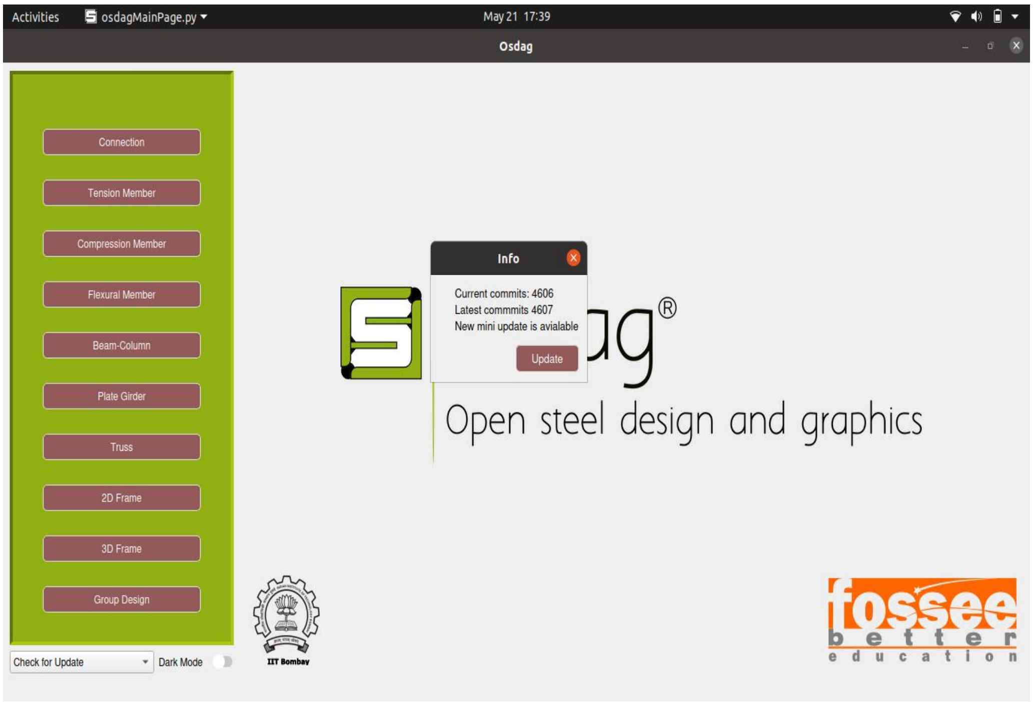
below are the screen shots of above feature



Info

New update is available at here

Major update found



Current commits: 4606
Latest commmits 4607
New mini update is avialable

Mini Update found

Note: Every time when some one is about to push code on Github he/she has to increment noOfCommits located in Osdag/_version.py by. For example current no of commits is 3400 in _version.py before pushing code to github. Then change this value to 3401 and then push code to github.

4.3 Create Backup for previous version

After installing new mini update osdag maintains backup of its previous version, in case new update has any bug then we can restore to previous version.

Chapter 5

Conclusion

On the whole, this internship was a useful experience. I have gained new knowledge, skills and met many new people. I achieved several learning goals, and have moved a step further in achieving other. I got insight into professional practice. Internship has proved to be satisfactory and it has allowed as an opportunity to get an exposure of the practical implementation of theoretical fundamentals. **Here** during the internship period I developed myskills in following softwares/tools :

1. Osdag
2. More On Python
3. Latex
4. Git and Github
5. Nsis Scripting

6. Shell Scripting
7. More on Ubuntu
8. Tools For Communication

Throughout the internship, I found that several things are important:

Critical and Analytical Thinking: To organize our tasks and assignment, we need to analyse our problemsand assignment, and to formulate a good solution tothe problem. We would have to set contingency planfor the solution, so that we are well prepared for the unforeseeable situations.

Time Management: As overall project staff and programmer are always racing against tight time line and packed schedule, a proper time management will minimize facing overdue deadlines. An effective time management allows us to do our assignment efficiently and meet our schedules. Scheduling avoids time wastage and allows us to plan ahead, and gaining more as a result.

Goal Management: It is better to sub-divide the goals to a few achievable tasks, so that we will be gain- ing more confidence by accomplishing those tasks

Tools for communication: Most of the open-source community uses the following tools for communicating, asking questions and announcements etc. While seeking solutions to some of the problems I learned to use these tools.

– **Mailing-list -** A mailing list is a collection of names and addresses used by an individual or an organization to send material to multiple recipients. The term is often extended to include the people subscribed to such a list, so the group of subscribers is referred to as ”the mailing list”, or simply ”the list”. Mailing lists preceded web fo- rums and can provide similar functionalities. I learned the difference between top-posting and bot- tom posting, and why the latter is preferred in the open-source community.

- **Forums -** An Internet forum, or message board, is an online discussion site where people can hold conversations in the form of posted messages. They differ from chat 22 rooms in that messages are often longer than one line of text, and are at least temporarily archived. I used to ask ubuntu, Launchpad forum, Unix stack exchange, learned about asking questions, providing detailed response etc. I would like to once again appreciate everyone who has made my internship training a superb experience.