



# FOSSEE Winter Internship Report

On

Development of a Working Installer for Osdag

Submitted by

**Mehendi Hasan**

*3rd Year B.Sc(H) Physics Major, Computer Science Minor*

*Kirori Mal College*

University of Delhi

Under the Guidance of

**Prof. Siddhartha Ghosh**

Department of Civil Engineering

Indian Institute of Technology Bombay

**Mentors:**

Ajmal Babu M S

Parth Karia

Ajinkya Dahale

February 9, 2025

# Acknowledgments

- I would like to express my heartfelt gratitude to everyone who supported and guided me throughout the course of this project. Their encouragement and assistance have been invaluable in helping me achieve my objectives and complete this work successfully.
- First and foremost, I extend my sincere thanks to the project staff of the Osdag team, including Ajmal Babu M. S., Ajinkya Dahale, and Parth Karia, for their unwavering support, insights, and expertise throughout the project. Their dedication and guidance have greatly enriched my learning experience.
- I am deeply grateful to Prof. Siddhartha Ghosh, Principal Investigator of the Osdag project and a faculty member in the Department of Civil Engineering at IIT Bombay, for his vision, mentorship, and encouragement. His leadership has been instrumental in shaping the success of this project.
- I would also like to acknowledge the invaluable support and guidance of Prof. Kanhan M. Moudgalya, Principal Investigator of the FOSSEE Project, Department of Chemical Engineering, IIT Bombay. His passion for fostering open-source initiatives has been a source of inspiration for me.
- My heartfelt thanks go to FOSSEE Managers Usha Viswanathan and Vineeta Parmar, as well as their entire team, for their constant support and coordination throughout the project. Their tireless efforts in facilitating the smooth execution of tasks were vital to the project's success.
- I am immensely thankful for the support provided by the National Mission on Education through Information and Communication Technology (ICT), Ministry

of Education (MoE), Government of India, whose initiative and resources made this project possible.

- I would also like to extend my gratitude to my colleagues who worked alongside me during this internship/project. Their collaboration, ideas, and camaraderie made the journey all the more enjoyable and productive.
- Finally, I am deeply appreciative of my college, my department, and my esteemed professors for their encouragement and support during my academic journey. I would also like to express my gratitude to the head of my department and the principal of my institution for providing me with the necessary resources and opportunities to excel in my endeavors.
- Thank you all for making this experience an enriching and rewarding one.

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	National Mission in Education through ICT . . . . .	5
1.1.1	ICT Initiatives of MoE . . . . .	6
1.2	FOSSEE Project . . . . .	7
1.2.1	Projects and Activities . . . . .	7
1.2.2	Fellowships . . . . .	7
1.3	Osdag Software . . . . .	8
1.3.1	Osdag GUI . . . . .	9
1.3.2	Features . . . . .	9
<b>2</b>	<b>Screening Task</b>	<b>10</b>
2.1	Problem Statement . . . . .	10
2.2	Tasks Done . . . . .	10
<b>3</b>	<b>Internship Task 1</b>	<b>14</b>
3.1	Task 1: Check Osdag Installation on Different OS . . . . .	14
3.2	Task 1: Tasks Done . . . . .	14
<b>4</b>	<b>Internship Task 2</b>	<b>15</b>
4.1	Task 2: Fix Missing Images in Osdag GUI . . . . .	15
4.2	Task 2: Tasks Done . . . . .	15
4.2.1	Explanation of the Code . . . . .	17
<b>5</b>	<b>Internship Task 3</b>	<b>18</b>
5.1	Task 3: Fix Missing Themes in Osdag GUI . . . . .	18
5.2	Task 3: Tasks Done . . . . .	18
<b>6</b>	<b>Internship Task 4</b>	<b>19</b>
6.1	Task 4: Missing Database Files . . . . .	19
6.2	Task 4: Tasks Done . . . . .	19
<b>7</b>	<b>Internship Task 5</b>	<b>20</b>

7.1	Task 5: Read on NSIS . . . . .	20
7.2	Task 5: Tasks Done . . . . .	20
<b>8</b>	<b>Internship Task 6</b>	<b>21</b>
8.1	Task 6: Create first version of Windows installer . . . . .	21
8.2	Task 6: Tasks Done . . . . .	21
8.2.1	Description of the Script . . . . .	21
8.2.2	Cleanup Temporary Files Section . . . . .	24
8.2.3	Uninstaller Section . . . . .	24
8.2.4	Code . . . . .	25
8.3	Task 6: Documentation . . . . .	33
8.3.1	Installation Guide . . . . .	33
<b>9</b>	<b>Conclusions</b>	<b>35</b>
9.1	Tasks Accomplished . . . . .	35
9.2	Skills Developed . . . . .	35
<b>A</b>	<b>Appendix</b>	<b>37</b>
A.1	Work Reports . . . . .	37
	<b>Bibliography</b>	<b>40</b>

# Chapter 1

## Introduction

### 1.1 National Mission in Education through ICT

The National Mission on Education through ICT (NMEICT) is a scheme under the Department of Higher Education, Ministry of Education, Government of India. It aims to leverage the potential of ICT to enhance teaching and learning in Higher Education Institutions in an anytime-anywhere mode.

The mission aligns with the three cardinal principles of the Education Policy—**access, equity, and quality**—by:

- Providing connectivity and affordable access devices for learners and institutions.
- Generating high-quality e-content free of cost.

NMEICT seeks to bridge the digital divide by empowering learners and teachers in urban and rural areas, fostering inclusivity in the knowledge economy. Key focus areas include:

- Development of e-learning pedagogies and virtual laboratories.
- Online testing, certification, and mentorship through accessible platforms like EduSAT and DTH.
- Training and empowering teachers to adopt ICT-based teaching methods.

For further details, visit the official website: [www.nmeict.ac.in](http://www.nmeict.ac.in).

### 1.1.1 ICT Initiatives of MoE

The Ministry of Education (MoE) has launched several ICT initiatives aimed at students, researchers, and institutions. The table below summarizes the key details:

No.	Resource	For Students/Researchers	For Institutions
<b>Audio-Video e-content</b>			
1	SWAYAM	Earn credit via online courses	Develop and host courses; accept credits
2	SWAYAMPBABHA	Access 24x7 TV programs	Enable SWAYAMPBABHA viewing facilities
<b>Digital Content Access</b>			
3	National Digital Library	Access e-content in multiple disciplines	List e-content; form NDL Clubs
4	e-PG Pathshala	Access free books and e-content	Host e-books
5	Shodhganga	Access Indian research theses	List institutional theses
6	e-ShodhSindhu	Access full-text e-resources	Access e-resources for institutions
<b>Hands-on Learning</b>			
7	e-Yantra	Hands-on embedded systems training	Create e-Yantra labs with IIT Bombay
8	FOSSEE	Volunteer for open-source software	Run labs with open-source software
9	Spoken Tutorial	Learn IT skills via tutorials	Provide self-learning IT content
10	Virtual Labs	Perform online experiments	Develop curriculum-based experiments
<b>E-Governance</b>			
11	SAMARTH ERP	Manage student lifecycle digitally	Enable institutional e-governance
<b>Tracking and Research Tools</b>			
12	VIDWAN	Register and access experts	Monitor faculty research outcomes
13	Shodh Shuddhi	Ensure plagiarism-free work	Improve research quality and reputation
14	Academic Bank of Credits	Store and transfer credits	Facilitate credit redemption

Table 1.1: Summary of ICT Initiatives by the Ministry of Education

## 1.2 FOSSEE Project

The FOSSEE (Free/Libre and Open Source Software for Education) project promotes the use of FLOSS tools in academia and research. It is part of the National Mission on Education through Information and Communication Technology (NMEICT), Ministry of Education (MoE), Government of India.

### 1.2.1 Projects and Activities

The FOSSEE Project supports the use of various FLOSS tools to enhance education and research. Key activities include:

- **Textbook Companion:** Porting solved examples from textbooks using FLOSS.
- **Lab Migration:** Facilitating the migration of proprietary labs to FLOSS alternatives.
- **Niche Software Activities:** Specialized activities to promote niche software tools.
- **Forums:** Providing a collaborative space for users.
- **Workshops and Conferences:** Organizing events to train and inform users.

### 1.2.2 Fellowships

FOSSEE offers various internship and fellowship opportunities for students:

- Winter Internship
- Summer Fellowship
- Semester-Long Internship

Students from any degree and academic stage can apply for these internships. Selection is based on the completion of screening tasks involving programming, scientific computing, or data collection that benefit the FLOSS community. These tasks are designed to be completed within a week.

For more details, visit the official FOSSEE website.





Figure 1.1: FOSSEE Projects and Activities

### 1.3 Osdag Software

Osdag (Open steel design and graphics) is a cross-platform, free/libre and open-source software designed for the detailing and design of steel structures based on the Indian Standard IS 800:2007. It allows users to design steel connections, members, and systems through an interactive graphical user interface (GUI) and provides 3D visualizations of designed components. The software enables easy export of CAD models to drafting tools for construction/fabrication drawings, with optimized designs following industry best practices [1, 2, 3]. Built on Python and several Python-based FLOSS tools (e.g., PyQt and PythonOCC), Osdag is licensed under the GNU Lesser General Public License (LGPL) Version 3.

### 1.3.1 Osdag GUI

The Osdag GUI is designed to be user-friendly and interactive. It consists of

- **Input Dock:** Collects and validates user inputs.
- **Output Dock:** Displays design results after validation.
- **CAD Window:** Displays the 3D CAD model, where users can pan, zoom, and rotate the design.
- **Message Log:** Shows errors, warnings, and suggestions based on design checks.

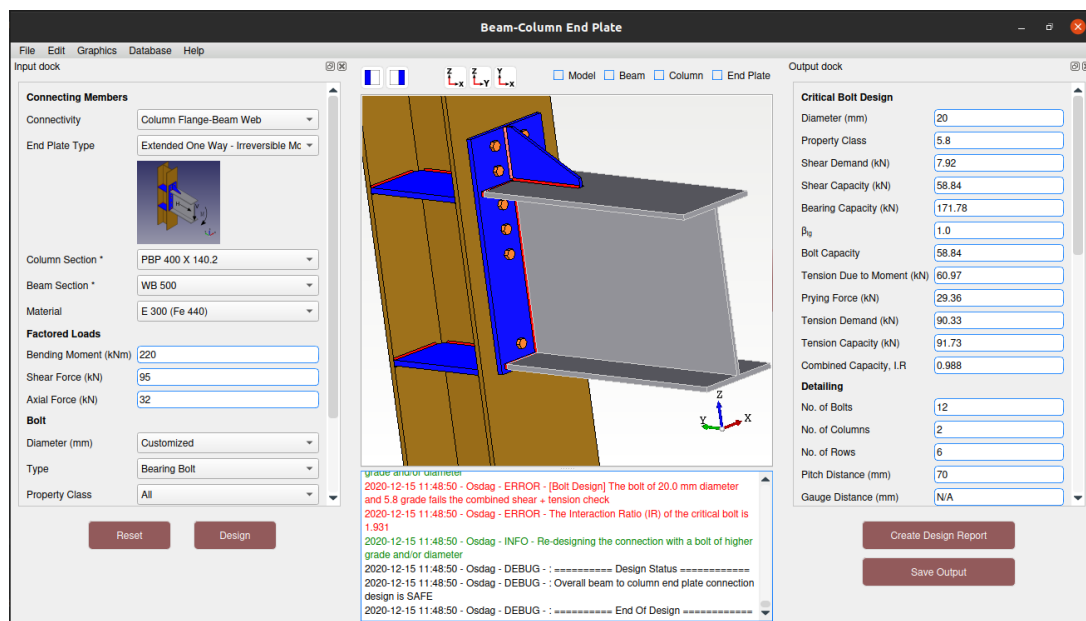


Figure 1.2: Osdag GUI

### 1.3.2 Features

- **CAD Model:** The 3D CAD model is color-coded and can be saved in multiple formats such as IGS, STL, and STEP.
- **Design Preferences:** Customizes the design process, with advanced users able to set preferences for bolts, welds, and detailing.
- **Design Report:** Creates a detailed report in PDF format, summarizing all checks, calculations, and design details, including any discrepancies.

For more details, visit the official Osdag website.

# Chapter 2

## Screening Task

### 2.1 Problem Statement

Package the provided prism viewer application using Conda

### 2.2 Tasks Done

- Setup environment for the Prism Viewer Application:

The Prism Viewer app relies on specific versions of Python packages, which vary based on version compatibility. To address this, I first identified these compatibility requirements and then created the conda environment for the app accordingly.

- Create a Conda recipe for the application:

After setting up the environment, I needed to package it using a conda recipe. This would allow other users to easily set up the application without worrying about version compatibility issues by simply creating the environment using this recipe.

Listing 2.1: Conda recipe

```
1 {% set version = "1.0.0" %}
2 {% set name = "prism_viewer" %}
3
4 package:
5     name: {{name}}
6     version: {{version}}
7
```

```

8 source:
9   path: ..\prism_viewer
10
11 build:
12   script: python setup.py install
13
14 requirements:
15   build:
16     - vs2015_runtime
17     - python=3.9
18     - conda-build
19     - libarchive
20     - setuptools
21   host:
22     - python=3.9
23   run:
24     - python=3.9
25     - pythonocc-core=7.8.1
26     - numpy
27     - pyqt=5.15
28     - PyQt5-sip=12.9
29
30 channels:
31   - conda-forge
32   - defaults

```

- Unit Test Application:

Unit test code was written to ensure that each component of the application functions correctly.

Listing 2.2: Unit Test Code

```

1 import unittest
2 from unittest.mock import patch, MagicMock
3 from PyQt5.QtWidgets import QApplication
4 import sys
5 from main import PrismViewer
6 from prism_calculator import PrismCalculator
7 app = QApplication(sys.argv)

```

```

8
9 class TestPrismCalculations(unittest.TestCase):
10     def setUp(self):
11         # Mock the database connection and cursor
12         with patch('sqlite3.connect') as mock_connect:
13             self.mock_conn = MagicMock()
14             self.mock_cursor = MagicMock()
15             self.mock_conn.cursor.return_value = self.mock_cursor
16             mock_connect.return_value = self.mock_conn
17
18         # Mock data to simulate the 'prisms' table with known
19             dimensions
20         mock_data = [
21             ('L01B01H01', 1.0, 1.0, 1.0),
22             ('L02B03H04', 2.0, 3.0, 4.0),
23             ('L10B20H30', 10.0, 20.0, 30.0)
24         ]
25         self.mock_cursor.fetchall.return_value = mock_data
26
27         # this use the mock data
28         self.viewer = PrismViewer()
29
30     def test_application_running(self):
31         # Test that the application initializes and runs without
32             issues
33         self.assertIsInstance(self.viewer, PrismViewer)
34
35         # Check if main window is visible
36         self.viewer.show()
37         self.assertTrue(self.viewer.isVisible())
38
39         # Check if UI components are present
40         self.assertIsNotNone(self.viewer.designation_dropdown)
41         self.assertIsNotNone(self.viewer.surface_area_label)
42         self.assertIsNotNone(self.viewer.volume_label)
43         self.assertIsNotNone(self.viewer.display_button)
44
45         #check if the dropdown is populated
46         self.assertGreater(self.viewer.designation_dropdown.count(),

```

```

    , 0)
45
46 def test_surface_area_and_volume_calculations(self):
47     # Dictionary with expected surface areas and volumes for
48     # each designation
49     expected_results = {
50         'L01B01H01': {'surface_area': 6.0, 'volume': 1.0},
51         'L02B03H04': {'surface_area': 52.0, 'volume': 24.0},
52         'L10B20H30': {'surface_area': 2200.0, 'volume': 6000.0}
53     }
54
55     # Iterate through each prism and check calculations
56     for designation, values in expected_results.items():
57         # Set the dropdown to the current designation
58         self.viewer.designation_dropdown.setCurrentText(
59             designation)
60
61         # Trigger the update display to calculate surface area
62         # and volume
63         self.viewer.update_display()
64
65         # Check the surface area and volume labels against
66         # expected values
67         self.assertEqual(self.viewer.surface_area_label.text(),
68             f"Surface Area: {values['surface_area']}")
69         self.assertEqual(self.viewer.volume_label.text(), f"
70             Volume: {values['volume']}")
71
72 if __name__ == '__main__':
73     unittest.main()

```

# Chapter 3

## Internship Task 1

### 3.1 Task 1: Check Osdag Installation on Different OS

A new method was developed to install Osdag using Conda. To ensure its integrity, it was essential to run extensive tests across various configurations and operating systems.

### 3.2 Task 1: Tasks Done

Installed Osdag on Windows 10, Windows 11 and, Ubuntu 22.04. Ran module checks if everything is working as expected.

# Chapter 4

## Internship Task 2

### 4.1 Task 2: Fix Missing Images in Osdag GUI

On installation with the new method the images in the Osdag GUI were missing.

### 4.2 Task 2: Tasks Done

Figured out why it was happening. The new method of installation using Conda was packaging non-Python files as resource files. These files were packaged alongside the application and could be accessed within the code. For example if `osdag.png` were to be referenced somewhere in the code, one needs to import them as follows

Listing 4.1: How to access resource files

```
1 import importlib.resources as files
2
3 osdag_png_path = files("osdag.data.ResourceFiles.images").joinpath("
    finplate.png"), 'Fin_Plate')
```

Now, all these references need to be updated to follow the new method of accessing files in the code. To streamline this process, I used an automation script to handle the necessary changes across the codebase. This script ensured that all file access references were replaced with the correct method of importing and accessing the resource files using `importlib.resources`

Listing 4.2: Automation Script



```

1 import os
2 import re
3
4 # Folder containing Python files
5 folder_path = r"C:\Users\1hasa\miniconda3\envs\osdag-by-conda\Lib\site-
   packages\osdag"
6
7 # Regular expression pattern to find the image path
8 pattern = r'"ResourceFiles/images/([^"]+)"|\''ResourceFiles/images/([^\'
   ]+)\''
9
10 # Replacement format
11 replacement_format = lambda image_name: f'str(files("osdag.data.
   ResourceFiles.images").joinpath("{image_name}"))'
12
13 def process_file(file_path):
14     # Read the file content
15     with open(file_path, 'r', encoding='utf-8') as f:
16         content = f.read()
17
18     # Find all matches for the pattern
19     matches = re.findall(pattern, content)
20
21     # If no matches, skip processing
22     if not matches:
23         return False
24
25     # # Extract image names
26     image_names = [match[0] or match[1] for match in matches]
27
28     # Display file name and occurrences
29     print(f"\nFile: {file_path}")
30     print("Occurrences:")
31     for image_name in image_names:
32         print(f" - {image_name}")
33
34     # Ask user for confirmation
35     confirm = input("Do you want to replace these occurrences in this
   file? (yes/no): ").strip().lower()

```

```

36     if confirm != 'yes':
37         return False
38
39     # Replace occurrences
40     def replacement_function(match):
41         image_name = match.group(1) or match.group(2)
42         return image_name
43
44     updated_content = re.sub(pattern, replacement_function, content)
45
46     # Write the updated content back to the file
47     with open(file_path, 'w', encoding='utf-8') as f:
48         f.write(updated_content)
49
50     print(f"Replacements made in {file_path}.")
51     return True
52
53 # Loop through all Python files in the directory and subdirectories
54 for root, dirs, files in os.walk(folder_path):
55     for file in files:
56         if file.endswith(".py"): # Process only Python files
57             file_path = os.path.join(root, file)
58             process_file(file_path)
59
60 print("Processing complete!")

```

## 4.2.1 Explanation of the Code

- **\*\*Line 1-2\*\***: Imports necessary libraries
- **\*\*Line 4-12\*\***: Define variables and pattern
- **\*\*Line 13-51\*\***: Define a function that processes each file, looks for matching pattern in the file and overwrites the file.
- **\*\*Line 52-58\*\***: Iterate through each file and pass it to the processfiles function.

# Chapter 5

## Internship Task 3

### 5.1 Task 3: Fix Missing Themes in Osdag GUI

On installation with the new method the Themes in the Osdag GUI were missing.

### 5.2 Task 3: Tasks Done

The problem was similar to the issue with the missing images. The themes were also packed with the Osdag application and needed to be accessed in the same manner as the other resource files. To resolve this, I applied the same approach used for accessing the non-Python files, ensuring the themes were imported and accessed correctly within the application. This ensured consistency and proper access to all bundled resources.

# Chapter 6

## Internship Task 4

### 6.1 Task 4: Missing Database Files

On installation with the new method the database files in the Osdag application were missing.

### 6.2 Task 4: Tasks Done

On the first-time installation of Osdag, the database files are created by running a .sqlite file, which initializes and adds data into the required tables. However, this .sqlite file was also a resource files, so it needed to be accessed accordingly, just like the other resources. To handle this, I updated the .sqlite file access logic to use the new method of importing the resource files, ensuring that the database initialization process works seamlessly by properly accessing the bundled resource files.

# Chapter 7

## Internship Task 5

### 7.1 Task 5: Read on NSIS

The Osdag application is currently packaged with Conda. To install Osdag, users need to know how to use Anaconda and install it from Conda channels. However, for general users who simply want to run Osdag and have no prior knowledge of Anaconda, a clean, user-friendly installer executable needs to be created. This installer would simplify the installation process, ensuring that Osdag can be easily set up without requiring any technical expertise related to Anaconda.

### 7.2 Task 5: Tasks Done

NSIS (Nullsoft Scriptable Install System) is an open-source, script-driven system for creating Windows installers. It is widely used due to its lightweight nature, flexibility, and ability to create professional and user-friendly installers.

- Read on NSIS, its scripting language and concepts
- How a windows installer can be created using NSIS for the Osdag application.
- Checked the universality of this installer for all windows users.

# Chapter 8

## Internship Task 6

### 8.1 Task 6: Create first version of Windows installer

### 8.2 Task 6: Tasks Done

Created first functional Windows installer of Osdag.

#### 8.2.1 Description of the Script

The script performs the following tasks:

##### General Information and Setup

- Defines Installer Name and Settings  
This sets the output file name to `osdag_installer.exe` and requires administrator privileges to run the installer.
- Includes Modern UI Library and Dialogs  
The Modern User Interface (MUI2) library is included to enhance the installer GUI.
- Define Installer Information  
This defines titles for the installer pages and enables an abort warning.
- Add Custom Icons and Header Image  
This sets a custom installer, uninstaller icons and a header image for the installer.

- **Add Installer Pages**  
Adds pages for welcome, license agreement, installation progress, and finish.

### **Miniconda Installation Section**

- **Extract Miniconda Installer**  
Extracts the Miniconda installer to the system's temporary directory.
- **Check if Miniconda is Already Installed**  
Asks the user whether Miniconda/Anaconda is already installed.
- **If Already Installed, Select Installation Folder**  
Allows the user to select the Miniconda installation folder if already installed.
- **If Not Installed, Install Miniconda**  
Installs Miniconda silently in the default user profile directory.
- **Verify Installation Path**  
Verifies the installation path or user-selected folder.

### **Osdag Installation Section**

- **Verify Conda Executable**  
Checks whether the conda executable is available in the specified path.
- **Create Conda Environment**  
Creates a new Conda environment named `osdag_env`
- **Install Osdag in Conda Environment**  
Installs the Osdag application within the created environment from conda channels.
- **Handle Errors**  
Displays an error message and aborts installation if `conda.exe` is not found.

### **MikTeX Installation Section**

- **Clear Any Existing Errors** Clears any previous errors that might interfere with the LaTeX installation process.

- **Copy MiKTeX Installer to Temporary Directory** Copies the MiKTeX installer executable to the system's temporary directory for use during installation.
- **Check for Existing LaTeX Installation** Executes the `where pdflatex` command to check if a LaTeX distribution (MiKTeX) is already installed on the system. The output is redirected to a temporary file.
- **Retrieve MiKTeX Path (if Installed)** If LaTeX is detected, extracts the installation directory of `pdflatex.exe` by parsing the command output.
- **Warn if LaTeX is Missing** Displays a warning message if LaTeX (`pdflatex`) is not found on the system, prompting the user to install it.
- **Install MiKTeX** Runs the MiKTeX installer, prompting the user to install for the current user without changing the default installation path.
- **Handle Installation Errors** Displays an error message and aborts the installation process if the MiKTeX installation fails due to system permissions or other issues.
- **Display Post-Installation Reminder** Informs the user to update MiKTeX before using Osdag to ensure all required LaTeX packages are installed and up-to-date.

### Create Desktop and Start Menu Shortcuts Section

- **Define Path for Desktop Shortcut** Sets the path where the desktop shortcut for Osdag will be created.
- **Copy Osdag Icon to Environment Directory** Copies the `Osdag_App.icon.ico` file to the appropriate directory inside the Osdag Conda environment, ensuring that the shortcut has a custom icon.
- **Create Desktop Shortcut** Creates a desktop shortcut named `Osdag.lnk`. The shortcut uses `cmd.exe` to activate the Conda environment and run the `osdag` application. It also assigns the custom icon to the shortcut.
- **Create Start Menu Shortcut** Creates a directory in the Start Menu under Programs Osdag and adds a shortcut to run Osdag in the same manner as the desktop shortcut. Assigns the custom icon to the Start Menu shortcut as well.



- **Add Uninstaller Shortcut** Adds an uninstaller shortcut (Uninstall.exe) to the Start Menu directory, allowing users to easily uninstall Osdag.
- **Add Osdag to Control Panel/Registry Keys** Creates registry entries for Osdag under HKCU Software Microsoft Windows CurrentVersion Uninstall:
  - **DisplayName:** Sets the name of the application to "Osdag" in the Programs and Features list.
  - **UninstallString:** Specifies the command to run the uninstaller ("\$SMPROGRAMS Osdag Uninstall.exe").
  - **InstallLocation:** Specifies the installation directory of Osdag's Conda environment.
  - **DisplayIcon:** Sets the icon for Osdag in the Programs and Features list.
  - **Publisher:** Sets the publisher to "Osdag".
  - **DisplayVersion:** Sets the application version to "1.0".
  - **NoModify or NoRepair:** Disables modify and repair options in the Programs and Features list.
- **Notify User of Shortcut Creation** Outputs a confirmation message to the user that the desktop and Start Menu shortcuts have been successfully created.

## 8.2.2 Cleanup Temporary Files Section

- **Delete Miniconda Installer**
- **Delete MikTeX Installer**
- **Delete Additional Temporary Files**

## 8.2.3 Uninstaller Section

- **Remove Conda Environment:** Reads the conda environment installation path from the registry and deletes the entire directory recursively using RMDir /r. This ensures that all files and folders related to the Osdag environment are removed.
- **Remove Application Shortcuts**

- Remove Uninstaller
- Remove Registry Keys

## 8.2.4 Code

Here's the detailed commented code of the NSIS(.nsi) script to generate an installer executable for Osdag,

Listing 8.1: NSIS Script Code

```

1 ; Define the output file name for the installer and set it to require
   admin privileges
2 OutFile "osdag_installer.exe"
3 RequestExecutionLevel admin
4
5 ; Include necessary libraries for Modern UI and dialogs
6 !include "MUI2.nsh" ; Include Modern UI 2 library for enhanced GUI
7 !include "nsDialogs.nsh" ; Include dialogs library for custom dialogs
8
9 ; Define installer information
10 !define MUI_WELCOMEPAGE_TITLE "This Setup will guide you through the
   installation of Osdag  $\r$\n$\r$\nIt will also install some python
   dependencies that are required to run Osdag$\r$\n $\r$\nPLEASE
   UNINSTALL ANY EARLIER VERSION OF OSDAG on your system before going
   ahead (See README.txt for reference)$\r$\n $\r$\nPlease click Next
   only after uninstalling the earlier version" ; Title for the welcome
   page
11 !define MUI_FINISHPAGE_TITLE "Thank You for Installing Osdag" ;
   Title for the finish page
12 !define MUI_ABORTWARNING ; Display a warning if the user
   tries to abort installation
13 !define MUI_ICON "Osdag.ico" ; Set a custom installer icon
14 !define MUI_UNICON "Osdag.ico" ; Set a custom uninstaller icon
15 !define MUI_HEADERIMAGE ; Enable a header image for the
   installer
16 !define MUI_HEADERIMAGE_BITMAP "Osdag_header.bmp" ; Set the header
   image file
17
18 ; Add Modern UI pages

```

```

19 !insertmacro MUI_PAGE_WELCOME           ; Welcome page
20 !insertmacro MUI_PAGE_LICENSE "license.txt" ; License agreement page
21 !insertmacro MUI_PAGE_INSTFILES        ; Installation progress page
22 !insertmacro MUI_PAGE_FINISH           ; Finish page
23
24 ; Set the installer language to English
25 !insertmacro MUI_LANGUAGE "English"
26
27 ; Define the installer name and branding text
28 Name "Osdag"
29
30 ; Declare variables for storing paths
31 Var /GLOBAL condaPath
32 Var /GLOBAL miktexPath
33 Var /GLOBAL env_name
34 Var /GLOBAL osdagIconPath
35 Var /GLOBAL osdagShortcutPath
36
37 ; Section to handle Miniconda installation
38 Section "Miniconda Installation"
39     ; Set the output path for temporary files
40     SetOutPath "$TEMP"
41
42     ; Copy the Miniconda installer to the temporary directory
43     File /oname=MinicondaInstaller.exe "C:\Users\lhasa\Downloads\
44         Miniconda3-latest-Windows-x86_64.exe"
45
46     ; Ask the user if Miniconda/Anaconda is already installed
47     MessageBox MB_YESNO|MB_ICONQUESTION "Is Miniconda/Anaconda already
48         installed on your system?" IDYES YesMiniconda IDNO NoMiniconda
49
50     YesMiniconda:
51         ; Create a dialog to let the user select the existing
52         ; installation folder
53         nsDialogs::Create
54         nsDialogs::SelectFolderDialog "Select the folder where
55             Miniconda/Anaconda is installed" "" $condaPath
56         Pop $condaPath
57         ${If} $condaPath == ""

```

```

54         ; Abort installation if no directory is selected
55         MessageBox MB_ICONEXCLAMATION "No directory selected.
           Installation will not continue."
56         Quit
57     ${EndIf}
58
59     ; Go to the section
60     Goto PathFound
61
62     NoMiniconda:
63         ; Create a dialog to let the user select the existing
           installation folder
64         nsDialogs::Create
65         nsDialogs::SelectFolderDialog "Select installational directory"
           "$PROFILE" $condaPath
66         Pop $condaPath
67         StrCpy $condaPath "$condaPath\Miniconda3"
68
69         DetailPrint "Installing Miniconda. It may take some time...",
70
71         ; Perform a silent installation of Miniconda
72         ExecWait "$TEMP\MinicondaInstaller.exe" /InstallationType=
           JustMe /AddToPath=1 /RegisterPython=0 /S /D=$condaPath'
73     ${If} ${Errors}
74         MessageBox MB_ICONSTOP "Error: Failed to install Miniconda.
           Please check the installer or your system permissions."
75         Quit
76     ${EndIf}
77     ; Go to the section
78     Goto PathFound
79
80     PathFound:
81         ; Print the detected or installed Miniconda path
82         DetailPrint "Miniconda Found at: $condaPath"
83 SectionEnd
84
85 ; Section to install Osdag using the Miniconda environment
86 Section "install osdag"
87     ; Print a message indicating the creation of a Conda environment

```

```

88     DetailPrint "Creating environment for osdag"
89     StrCpy $1 "$condaPath\Scripts\conda.exe" ; Path to the Conda
        executable
90
91     ${If} ${FileExists} "$1"
92         ; Assign a name for the Conda environment
93         StrCpy $env_name "osdag_env"
94
95         ; Create the Conda environment
96         nsExec::ExecToLog 'cmd.exe /C "$1" create -y -n $env_name'
97
98         ; Install Osdag in the created Conda environment
99         DetailPrint "Installing osdag..."
100        nsExec::ExecToLog 'cmd.exe /C "$1" install -n $env_name -y
            osdag::osdag'
101
102    ${Else}
103        ; Display an error message if Conda executable is not found
104        MessageBox MB_ICONSTOP "Error: Conda executable not found at $1
            . Please check the path."
105        Quit
106    ${EndIf}
107
108 SectionEnd
109
110 Section "LaTeX Installation"
111     ; Clear any existing errors
112     ClearErrors
113
114     ; Copy the MikTeX installer to the temporary directory
115     SetOutPath $TEMP
116     File /oname=MikTeX.exe "C:\Users\1hasa\Downloads\basic-miktex-24.1-
        x64.exe"
117
118     ; Define a temporary file to store the output
119     SetOutPath $TEMP
120     FileOpen $1 "$TEMP\pdflatex_check.txt" w
121     FileClose $1
122

```

```

123 ; Run the "where pdflatex" command and redirect output to the file
124 ExecWait 'cmd.exe /C "where pdflatex > $TEMP\pdflatex_check.txt"'
125
126 ; Read the output from the file
127 FileOpen $1 "$TEMP\pdflatex_check.txt" r
128 FileRead $1 $miktexPath
129 FileClose $1
130
131
132 ${If} $miktexPath == ""
133     Goto install
134
135 ${Else}
136     ; Retrieve Latex installation directory
137     StrLen $R0 $miktexPath ; Get the length of the full string
138
139     ; Find the position of "\condabin\conda.bat"
140     StrCpy $R1 "\miktex\bin\x64\pdflatex.exe"
141     StrLen $R2 $R1 ; Length of "\condabin\conda.bat"
142
143     ; Subtract 1 to avoid including the trailing backslash before
144     condabin
145     IntOp $R3 $R0 - $R2
146     IntOp $R3 $R3 - 2 ; Subtract 1 more to exclude the last
147     backslash before condabin
148
149     ; Copy everything before "\condabin\conda.bat"
150     StrCpy $miktexPath $miktexPath $R3
151
152     DetailPrint "LaTeX found at: $miktexPath"
153     Goto End
154 ${EndIf}
155
156 install:
157     MessageBox MB_ICONEXCLAMATION "LaTeX not found (pdflatex is
158     missing). Please install MikTeX before continuing."
159
160     ; Run the MikTeX installer silently
161     DetailPrint "Installing MikTeX, please wait..."

```

```

159     MessageBox MB_ICONEXCLAMATION "Install for Current User. Do not
        change the default installation path for MikTeX."
160     ExecWait "$TEMP\MikTeX.exe"
161     ${If} ${Errors}
162         MessageBox MB_ICONSTOP "Error: Failed to install Miniconda.
        Please check the installer or your system permissions."
163         Quit
164     ${EndIf}
165
166     ; Run the "where pdflatex" command and redirect output to the
        file
167     StrCpy $miktexPath "$PROFILE\AppData\Local\Programs\MikTeX\"
168     DetailPrint "MikTeX Installed at $miktexPath"
169     MessageBox MB_ICONEXCLAMATION "Make sure to check updates for
        MikTeX before launching Osdag"
170
171     Goto End
172 End:
173 SectionEnd
174
175 ; Section to create shortcuts for Osdag
176 Section "Create Desktop and Start Menu Shortcuts"
177     ; Path for the desktop shortcut
178     StrCpy $osdagShortcutPath "$DESKTOP\Osdag.lnk"
179
180     SetOutPath $TEMP
181     File /oname=Osdag_App_icon.ico "C:\Users\lhasa\Osdag\installer\
        Osdag_App_icon.ico"
182
183     CopyFiles "$TEMP\Osdag_App_icon.ico" "$condaPath\envs\$env_name\Lib
        \site-packages\osdag\data\ResourceFiles\images"
184     StrCpy $osdagIconPath "$condaPath\envs\$env_name\Lib\site-packages\
        osdag\data\ResourceFiles\images\Osdag_App_icon.ico"
185
186     ; Create a desktop shortcut for Osdag
187     DetailPrint "Creating Desktop Shortcut for Osdag..."
188     CreateShortcut "$osdagShortcutPath" "$SYSDIR\cmd.exe" "/C call
        $condaPath\Scripts\activate.bat $env_name && osdag" "
        $osdagIconPath"

```

```

189
190 ; Create a Start Menu shortcut for Osdag
191 DetailPrint "Creating Start Menu Shortcut for Osdag..."
192 CreateDirectory "$SMPROGRAMS\Osdag"
193 CreateShortcut "$SMPROGRAMS\Osdag\Osdag.lnk" "$SYSDIR\cmd.exe" "/C
    call $condaPath\Scripts\activate.bat $env_name && osdag" "
    $osdagIconPath"
194
195 ; Add uninstaller script
196 WriteUninstaller "$SMPROGRAMS\Osdag\Uninstall.exe"
197
198 # Add to Control Panel/Registry Keys
199 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "DisplayName" "Osdag"
200 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "UninstallString" "$SMPROGRAMS\Osdag\Uninstall.
    exe"
201 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "InstallLocation" "$condaPath\envs\$env_name"
202 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "DisplayIcon" $osdagIconPath
203
204 ; Need to be confirmed
205 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "Publisher" "Osdag"
206 WriteRegStr HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "DisplayVersion" "1.0"
207 WriteRegDWORD HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "NoModify" 1
208 WriteRegDWORD HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag" "NoRepair" 1
209
210 ; Notify the user that the shortcuts have been created
211 DetailPrint "Desktop and Start Menu shortcuts for Osdag have been
    created."
212 SectionEnd
213
214
215 Section "Cleanup Temporary Files"

```



```

216     DetailPrint "Cleaning up temporary files..."
217
218     ; Delete Miniconda installer
219     Delete "$TEMP\MinicondaInstaller.exe"
220     ${If} ${FileExists} "$TEMP\MinicondaInstaller.exe"
221         DetailPrint "Failed to delete MinicondaInstaller.exe"
222     ${Else}
223         DetailPrint "Deleted MinicondaInstaller.exe"
224     ${EndIf}
225
226     ; Delete MikTeX installer
227     Delete "$TEMP\MiKTeX.exe"
228     ${If} ${FileExists} "$TEMP\MiKTeX.exe"
229         DetailPrint "Failed to delete MiKTeX.exe"
230     ${Else}
231         DetailPrint "Deleted MiKTeX.exe"
232     ${EndIf}
233
234     ; Delete any other temporary files
235     Delete "$TEMP\pdflatex_check.txt"
236     Delete "$TEMP\Osdag_App_icon.ico"
237
238
239
240     DetailPrint "Temporary files cleanup completed."
241 SectionEnd
242
243
244
245 ; Uninstaller Section
246 Section "Uninstall"
247
248     ; remove osdag conda environment
249     Var /GLOBAL condaEnvPath
250     ReadRegStr $condaEnvPath HKCU "Software\Microsoft\Windows\
        CurrentVersion\Uninstall\Osdag" "InstallLocation"
251     RMDir /r "$condaEnvPath"
252
253     ; remove app shortcuts

```

```

254 Delete "$DESKTOP\Osdag.lnk"
255 Delete "$SMPROGRAMS\Osdag\Osdag.lnk"
256
257 ; remove uninstaller
258 Delete "$SMPROGRAMS\Osdag\Uninstall.exe"
259 RMDir /r "$SMPROGRAMS\Osdag"
260
261 # Remove registry keys
262 DeleteRegKey HKCU "Software\Microsoft\Windows\CurrentVersion\
    Uninstall\Osdag"
263
264 MessageBox MB_OK "Osdag Uninstalled. You can remove MikTeX and Conda
    manually"
265
266 SectionEnd

```

## 8.3 Task 6: Documentation

Insert your contribution towards Osdag Developer/ User manual.

The NSIS script generates an executable file (.exe) for the Windows operating system. This executable is distributed to users for installing Osdag.

### 8.3.1 Installation Guide

- Double-click on the osdag\_installer.exe file to begin the installation process.
- A pop-up will appear requesting administrative privileges. Click "Yes" to proceed.
- The installation wizard will open. Accept the license agreement and click "Next."
- A dialog box will appear asking if Anaconda or Miniconda is already installed on your system: If Yes, select the folder where Anaconda/Miniconda is installed (usually Miniconda3 or Anaconda3). If No, Miniconda will be installed in the default path (UserProfile).
- After setting up Miniconda, the installer will create a Conda environment and install Osdag within this environment.

- Upon successful installation, a desktop shortcut and a Start Menu shortcut will be created, allowing you to launch Osdag with a single click.

# Chapter 9

## Conclusions

### 9.1 Tasks Accomplished

During the initial days of my internship, I focused on understanding the Osdag codebase, exploring how the components were interconnected, and gaining insights into the application's structure and growth. I resolved several bugs that occurred during new installations, such as missing database files, images, and themes. Following this, I transitioned to developing a Windows installer and successfully created the first functional installer for Osdag.

### 9.2 Skills Developed

During the fellowship, I gained valuable technical and professional skills, including:

- **Conda Packaging:** I learned how to effectively use Conda for environment management and packaging, allowing me to automate the installation of dependencies and manage isolated environments for different applications.
- **Python Scripting and Debugging:** I developed my skills in Python scripting, writing efficient code for automation and installation tasks. Additionally, I honed my debugging abilities by identifying and resolving issues in the Osdag application, ensuring a smooth installation process.
- **NSIS Installer Creation:** I gained hands-on experience with NSIS (Nullsoft Scriptable Install System) to create a functional Windows installer, learning how to

customize installation workflows, handle user inputs, and manage file installations efficiently. These experiences have significantly enhanced my technical abilities and provided valuable insights into software deployment and environment management.

# Chapter A


## Appendix

### A.1 Work Reports

OSDAG Task DataSheet

 DATE	DAY	TASK	#	Hours worked
13 Nov 2024	Wednesday	installed OSDAG on windows 10 x64 machine and		2
14 Nov 2024	Thursday	installed OSDAG on windows 10 x64 machine and		1
15 Nov 2024	Friday	Design Reports Error (Report was not being genera		3
16 Nov 2024	Saturday	Installed OSDAG on windows 11 and Ubuntu 22.04		4
18 Nov 2024	Monday	Missing Images In osdag application GUI		3
19 Nov 2024	Tuesday	Studied on including and referencing non-python fil		2
20 Nov 2024	Wednesday	Studied on including and referencing non-python fil		2
21 Nov 2024	Thursday	Tried to fix missing images in the GUI		3
22 Nov 2024	Friday	Found problem in referencing resourceFiles, Chang		4
23 Nov 2024	Saturday	Looked into themes not found issue, Fixed the issu		4
24 Nov 2024	Sunday	Installed osdag again with latest changes/fix in Wir		2
25 Nov 2024	Monday	Looked into .sqlite files(weather to have them with		3
26 Nov 2024	Tuesday	Addressed PR comments and changes, Installed os		3
27 Nov 2024	Wednesday	looked into no images in design report( in windows		2
30 Nov 2024	Saturday	Reading on NSIS Installer		2
2 Dec 2024	Monday	Reading on NSIS Installer		3
3 Dec 2024	Tuesday	Reading on NSIS Installer		2
9 Dec 2024	Monday	Attempted to make an installer for windows		4.5
10 Dec 2024	Tuesday	Attempted to make an installer for windows		4
16 Dec 2024	Monday	Trying a different approach for installation of Minic		5
19 Dec 2024	Thursday	Trying a different approach for installation of Minic		2
20 Dec 2024	Friday	Trying a different approach for installation of Minic		2
21 Dec 2024	Saturday	Created the first functional insatller that installs mi		3.5
25 Dec 2024	Wednesday	Made some changes in the initial installer		2
26 Dec 2024	Thursday	Added desktop and start menu icons for osdag		2
27 Dec 2024	Friday	Made installer with Modern UI for better user exper		3
28 Dec 2024	Saturday	Created Internship report		4
1 Jan 2025	Wednesday	Added Miktex Installation		4
2 Jan 2025	Thursday	Updated first version installer		1
3 Jan 2025	Friday	Updated first version installer		3
4 Jan 2025	Saturday	Added osdag to control panel and added registry ke		2
5 Jan 2025	Sunday	Added Uninstaller for Osdag		4

OSDAG Task DataSheet

	DATE	DAY	TASK	#	Hours worked
	6 Jan 2025	Monday	Remove temp files on installation		3



# Bibliography

- [1] Siddhartha Ghosh, Danish Ansari, Ajmal Babu Mahasrankintakam, Dharma Teja Nuli, Reshma Konjari, M. Swathi, and Subhrajit Dutta. Osdag: A Software for Structural Steel Design Using IS 800:2007. In Sondipon Adhikari, Anjan Dutta, and Satyabrata Choudhury, editors, *Advances in Structural Technologies*, volume 81 of *Lecture Notes in Civil Engineering*, pages 219–231, Singapore, 2021. Springer Singapore.
- [2] FOSSEE Project. FOSSEE News - January 2018, vol 1 issue 3. Accessed: 2024-12-05.
- [3] FOSSEE Project. Osdag website. Accessed: 2024-12-05.