

Detailed Description of the Application Building Process

Contents

1.Introduction -----	1
2.Installing Required Package -----	1
3.Creating ExecutableFile(Exe) -----	1
4.Creating Installer -----	2

Introduction

Here we will be building a python application where original images and filtered images are shown. Then making an executable file(exe) from the python file. Later we will make an installer(set up program) which makes it runnable in any computer from the exe file.

Installing Required Package

We will need to use some python package. That should be installed before we start working on building the project. Here a glimpse of all the things we need to install.

1. Installing Pillow

We install Pillow 8.4.0 using `pip3 install Pillow` this command. The Python Imaging Library adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It should provide a solid foundation for a general image processing tool.

2. Installing PyInstaller

We install pyinstaller 5.2 using `pip install pyinstaller` this command. PyInstaller bundles a Python application and all its dependencies into a single package. The user can run the packaged app without installing a Python interpreter or any modules. PyInstaller supports Python 3.7 and newer, and correctly bundles many major Python packages such as numpy, matplotlib, PyQt, wxPython, and others.

3. Tkinter

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Here we import Tkinter on our file to use it.

```
from tkinter import *  
import tkinter as tk
```

4. Creating ExecutableFile(Exe)

To create an executable file from a python file we use these command

```
C:\Windows\System32\cmd.exe
```

```
Microsoft Windows [Version 10.0.19044.1826]
```

```
(c) Microsoft Corporation. All rights reserved.
```

```
D:\Fourth Year First Semester\DIP Lab\Tkinter\New folder>pip install pyinstaller
Requirement already satisfied: pyinstaller in c:\users\user\appdata\local\programs\py
Requirement already satisfied: pyinstaller-hooks-contrib>=2021.4 in c:\users\user\app
Requirement already satisfied: altgraph in c:\users\user\appdata\local\programs\pytho
Requirement already satisfied: setuptools in c:\users\user\appdata\local\programs\pyt
Requirement already satisfied: pywin32-ctypes>=0.2.0 in c:\users\user\appdata\local\p
Requirement already satisfied: pefile>=2022.5.30 in c:\users\user\appdata\local\progr
Requirement already satisfied: future in c:\users\user\appdata\local\programs\python\
```

```
D:\Fourth Year First Semester\DIP Lab\Tkinter\New folder>pyinstaller --onefile -w DIP_App.py
141 INFO: PyInstaller: 5.2
141 INFO: Python: 3.9.5
155 INFO: Platform: Windows-10-10.0.19044-SP0
156 INFO: wrote D:\Fourth Year First Semester\DIP Lab\Tkinter\New folder\DIP_App.spec
160 INFO: UPX is not available.
170 INFO: Extending PYTHONPATH with paths
['D:\Fourth Year First Semester\\DIP Lab\\Tkinter\\New folder']
872 INFO: checking Analysis
873 INFO: Building Analysis because Analysis-00.toc is non existent
874 INFO: Initializing module dependency graph...
881 INFO: Caching module graph hooks...
956 INFO: Analyzing base_library.zip ...
5163 INFO: Processing pre-find module path hook distutils from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python
_find_module_path\\hook-distutils.py'.
5166 INFO: distutils: retargeting to non-venv dir 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python39\\l
10800 INFO: Caching module dependency graph...
10987 INFO: running Analysis Analysis-00.toc
11003 INFO: Adding Microsoft.Windows.Common-Controls to dependent assemblies of final executable
required by C:\Users\User\AppData\Local\Programs\Python\Python39\python.exe
11097 INFO: Analyzing D:\Fourth Year First Semester\DIP Lab\Tkinter\New folder\DIP_App.py
13190 INFO: Processing pre-find module path hook site from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Py
d_module_path\\hook-site.py'.
13206 INFO: site: retargeting to fake-dir 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python39\\lib\\site
20021 INFO: Processing module hooks...
20023 INFO: Loading module hook 'hook-difflib.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python
20031 INFO: Loading module hook 'hook-distutils.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Pyth
20033 INFO: Loading module hook 'hook-distutils.util.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\
20037 INFO: Loading module hook 'hook-encodings.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Pyth
20519 INFO: Loading module hook 'hook-heapq.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python39
20526 INFO: Loading module hook 'hook-lib2to3.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python
20627 INFO: Loading module hook 'hook-multiprocessing.util.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\P
'...
20633 INFO: Loading module hook 'hook-numpy.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Python\\Python39
20812 INFO: Import to be excluded not found: 'f2py'
20843 INFO: Loading module hook 'hook-numpy._pytesttester.py' from 'C:\\Users\\User\\AppData\\Local\\Programs\\Py
```

5. Creating Installer from Exe

Last step but not the least, we make the installer file from the executable file. Here, **NSIS** software are used to make the installer. Firstly we download the software application.



Then use it making installer.

