No File Found

Software Deployment Plan

Latest update: 2023-11-12

1. System Requirements

Support operating system:

- Windows 10 and above
- Linux SuSE Leap 15 and above
- Rocky 8
- Ubuntu14 and above
- Fedora 37 and above

Required dependencies:

- Windows
  - none
- Linux
  - Python version 3.11.4 or older

2. Deployment Strategy Summary

Our software will be deployed as a standalone application written in PyQT6. We will make use of the PyInstaller tool to package our application into executable files for each supported operating system. The final distribution will be available as:

- An .exe file for Windows.
- A binary executable file for linux

3. Installation Package Contents

3.1 Required source or compiled files

- no_file_found.exe
- no_file_found

3.2 Required third-party components

- PyQT6 runtime libraries

3.3 Required graphical assets, configuration and other non-program files

- Application icons and imagery

- Default configuration file (if any)
- Theme or skin files (if applicable)

3.4 Documentation files to be provided

- File Finder Installation Instruction.pdf : Contains information and steps for downloading and installing for supported operating systems
- User Manual.pdf : contains information regarding the usage of the application

3.5 Development files and components that must be excluded

- Any debugging tools or logs
- Source code (unless open source)
- Test files and test data
- py cache

4. Additional Code Required for Deployment

-

5. Deployment Tasks

1. Compile and test the PyQT6 application in a development environment.
2. Use PyInstaller to create executables for each OS.
   For Windows:
   Use PyInstaller on the main script to automatically generate a single .exe file with all the dependencies included
   For Linux:
   Use PyInstaller on the main script to automatically generate a single binary, executable file.
3. Ensure all necessary assets and files are bundled.
4. Write/update installation instruction file.
5. Conduct internal testing of the installation packages on supported operating systems.
6. Fix any identified bugs or issues.
7. Finalize the installation packages.
8. Distribute the installation packages to users via a github release.

6. Deployment Test Plan

1. Installation Test: Ensure that the software installs correctly without any errors on all supported operating systems.
2. Functional Test: Verify that all functionalities of the software work as expected on different platforms.
3. Uninstallation Test: Check that the software uninstalls completely..