

py2py

--

Programming Assignment 3, Project B

Peer-to-Peer File Sharing

CSE 4344

Spring 2015

Christina Dan
Stephen Shelton
Richard Victor

TABLE OF CONTENTS

1. Objective.....	2
2. Implementation	2
3. Challenges.....	2
4. Known Issues & Limitations.....	3
5. User Guide	3

OBJECTIVE

“Develop a peer to peer (P2P) file sharing software program which is used to search for other connected computers (peers) on the network to locate desired content.”

IMPLEMENTATION

For our application, we found a helpful tutorial online¹ explaining at a high-level how a peer-to-peer application is implemented. We chose to use Python v2.7 in our implementation because it is stable and is the version that we are most comfortable with. For the GUI, we decided to use the Qt² application framework with PyQt5³ because of its ease of use for designing cross-platform GUIs. Because our group consisted of Windows, Mac, and Linux users, we felt it was important our application could be developed on any of those platforms.

In order to create a productive development process within our group, we set up a repository on Bitbucket and used Git for version control so that we could individually work on the project as needed. Since only one of us has experience with Git via command line, we decided to use SourceTree⁴, a user-friendly Git desktop client.

We used PyInstaller⁵ to package our project into .exe and .app so our project could be used without having to build from source.

CHALLENGES

One of the problems we faced initially was installing PyQt5 with Python v2. By default, PyQt5 installs with Python 3, so we had to go through extra steps in order to make it work. Since we developed our application in both Mac OS X and Windows environments, each platform faced different challenges with installing PyQt5.

When testing our application, we discovered that there were a couple of problems transferring large (~100MB) files between peers. We remedied this by setting a hard cap of 25 MB for file size. Another issue we had with file transferring was that our main application would lock up while transferring files. This was fixed by handling file transfers in a separate thread in the background.

¹ <http://cs.berry.edu/~nhamid/p2p/index.html>

² <http://www.qt.io>

³ <http://www.riverbankcomputing.com/software/pyqt/intro>

⁴ <http://www.sourcetreeapp.com>

⁵ <https://github.com/pyinstaller/pyinstaller/wiki>

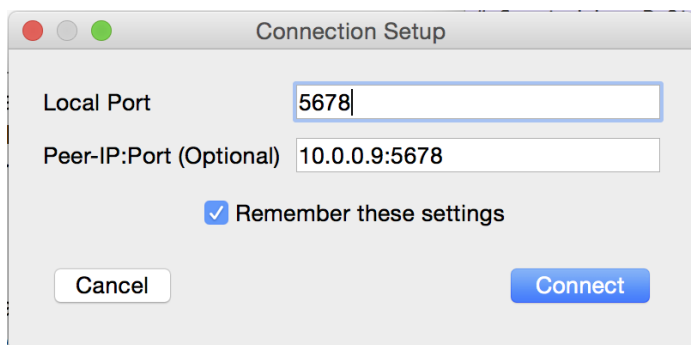
Because of all the troubles we faced installing all the frameworks necessary to develop our application, we decided to package our project as an executable so that anyone trying to run our application would not have to go through the trouble of installing everything necessary to build from source. Although PyInstaller worked without a hitch on Mac OS X, we ran into some issues with it not including all necessary modules in the final build. We ultimately resolved this by copying all the PyQt5 libraries into the executable directory.

KNOWN ISSUES & LIMITATIONS

- Maximum file size is 25 MB.
- There are some issues with the graphical peer list not always being updated when new peers join.

USER GUIDE

1. Run the application. For instructions on how to run our application, please refer to “README.txt”.
2. After starting the application, you should see the following dialog:

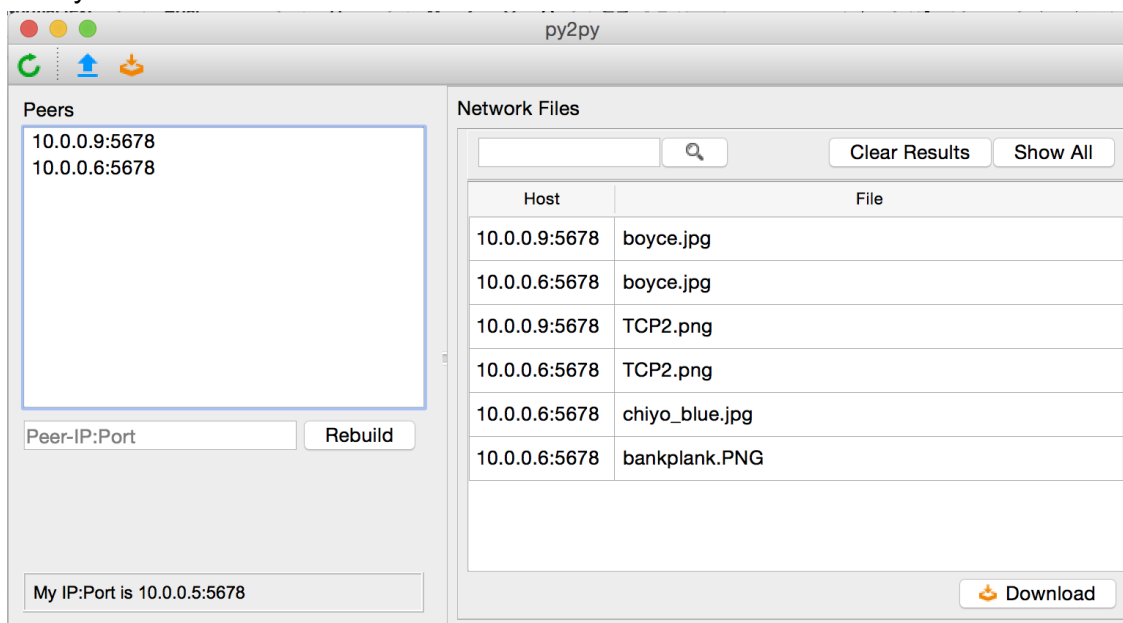


For “Local Port”, specify the port for which you want your server to be accessed. If none is provided, a default port of 5678 will be used.


Enter the IP address and port and a known peer if you are trying to join an existing network. **If an invalid peer IP and port (or none) is provided, you will create your own network.**


If “Remember these settings” is checked, the application will remember your inputs the next time it is started.

3. Next you'll be taken to the main window:



Two lists, one for peers and one for files currently on the network, will be displayed.

Upload a file: Click  (on some platforms, this may appear as “Upload”). Select a file from the file browser that appears and click “Upload” to begin hosting your file on the network.

Download a file: Select a file from the file list on the right, then click  or the “Download” button below the file list.

Refresh your lists: Click  (on some platforms, this may appear as “Refresh”).

Search for a file on the network: Enter a keyword (case sensitive and substrings) for a file you wish to find on the network. If a match is found, it will be displayed in your file list.

Clear search results: Click “Clear results”. *Note: Local files (e.g. files you have uploaded) will still be displayed.*

Show all files hosted on the network: Click “Show All”.

Rebuild your network: Enter a peer’s *IP:Port* and click “Rebuild”. This will rebuild your peer list from all peers known by given *IP:Port*.