

Eylül Taşyürek

2007100706

Seçkin Savaşçı

2008400078

---

QT 4.7.0 and QT Creator 2.0 is used. (placeholder is not included in older versions.)

---

This project aims to create an FTP Client by using QT. This client performs basic upload/download operations. We used QFtp class for network operations in the project.

During development, we divided our tasks into four groups. At first, we designed the user interface for Ftp Client. Instead of using list implementation, we decided to use treeview implementation. Although it was harder to implement the treeview, it is more elegant and professional looking. In our design, we avoided to get data from an “open” button, instead we showed entire directory with treeview. We did not use labels for “name”, “password”, and “hostname”, we preferred to use placeholder text of linedit objects. We also added a progress bar to inform user about sending and receiving processes. Then, the design includes a label line for further information about processes at the right bottom corner.

At second step, we build our treeview objects for local and remote server. We used model structure for this implementation. For local view, there was an available model for our needs but the situation was not the same for remote server. For implementing the treeview of the remote server, we handcoded an entirely new custom model called “FtpModel”. This implementation took 2/3 of whole code. Also we embedded sending and receiving operations in this model. So it is safe to use our new model to any application.

At third step, we developed and connected our core structures, including view objects, buttons, models, ftp operations... For this part, we tried to follow self-documented code paradigm.

At fourth step, we focused on error-prone situations. Mainly, we avoided the closing of the program by accident, so we overloaded closeevent slot. We informed user about overwriting a file and privilege needed transfers such as copying a file into a partition’s root directory (Windows).

---

Key features of this Ftp Client:

- Treeview is used instead of listview. – professional looking
- “open” button is not used. – multiple selection of files in different directories.
- An entirely new model is created for viewing remote server – professional looking and generic usage.
- Pressing “close” button (X) asks user for exit to avoid accidental closure of program.
- User is informed for overwriting situations, eg. Same named file in destination directory.
- User is informed if program cannot progress due to restrictions and privileges.
- User informed for error occurrences via interface.