Название проекта: Мессенджер на SSH

Автор проекта: Гарагуля Артур

Описание идеи:

Идея проекта состоит в реализации удобного общения пользователей внутри локальной сети без выхода в интернет. Программа должна работать как на одном компьютере, так и между пользователями разных. Для передачи сообщений используется протокол SSH (Secure Shell). В качестве интерфейса используется как и графический (QT), так и командный (CLI).

Описание реализации:

Мессенджер написан на языке Python и для общения использует утилиты работы с SSH Linux.

Реализован класс ‘Message’, хранящий только данные сообщения (за неимением структур как в С++, используется класс).

Объекты сообщения для передачи временно хранятся в разобранном виде в файле .pkl (pickle lib)

В базе данных хранятся только данные о текущем диалоге с пользователем. Изначально таблица пуста, но пополняется по мере добавления друзей. Формат таблицы – .db.

Для обработки принятых сообщений используется дополнительный поток (threading lib в CLI версии, класс, основанный на QThread в QT версии).

Для удобного получения и добавления данных в таблицы реализован класс ‘Database’.

Для добавления пользователей используется диалоговое окно и класс ‘AddFriendDialog’.

Для работы с основным окном и взаимодействием с пользователем реализован класс ‘Window’. Так как он является наследником стандартного основного окна, то имеет меню, необходимое для работы с видом программы, настройками, etc. При помощи QLabel пользователь получает данные о себе.

Обмен сообщениями происходит на основе терминальной утилиты Linux, называемой scp. Её работа аналогичка работе cp, но к пути назначения добавляются данные о пользователе SSH. В моём случае происходит синхронизация содержимого бинарных файлов ‘inmes.pkl’ и ‘outmes.pkl’. Класс ‘MessageChecker’ узнаёт, если изменяется последняя дата редакции файла ‘inmes.pkl’ и передаёт сообщение пользователю.

Для более приятного вида интерфейса используется стиль ‘breeze’. Он добавляется в программу при помощи импорта .qss файла.

Во время написания кода использовались аннотации.

Для запуска приложения используется .sh-скрипт. Тем не менее, сохраняется возможность запускать .py-файл, предварительно выдав ему права на исполнение (chmod +x window.py && ./window.py | python3 window.py)

Программа самостоятельно определяет пути до своих директорий и может быть запущена откуда угодно, работая только с файлами своей директории.

Необходимые библиотеки:

Помимо стандартных библиотек, пользователю необходимы:

1. pyqt5

2. sqlite3

3. pyperclip

4. pickle

Установка необходимых библиотек осуществляется скриптом ‘install.sh’ (./install.sh), использующим pip3 по пути /*usr*/bin/pip3.

Скриншоты:



