# 1. Введение

1.1 Цель

Цель проекта – создание десктопного приложения для ОС на базе ядра Linux, служащего для обеспечения обмена сообщениями внутри локальной сети или одного компьютера между пользоваетлями.

1.2 Возможности

В функционал приложения входит обеспечение удобного пользовательского интерфейса для обмена сообщениями по SSH. Функции работы с SHH представлены системными командами ОС. Программа умеет работать с базой данных. Для нормальной работы программа использует двухпоточность.

1.3 Ссылки

Интерфейс приложения будет ссылаться на интерфейс всем знакомых мессенджеров. Идея обмена сообщениями по средствам синхронизации содержимого бинарных файлов сообщений пренадлежит автору.

1.4 Краткое содержание

Данное приложение будет сохранять пользовательское сообщение в виде файла .pkl, хранящего в себе разобранный объекс класса, содержащий данные о сообщении. Для ведения диалога сообщения будут сохраняться в базу данных с указанием необходимых данных. Интерфейс будет доступен как в GUI (Qt), так и в CLI (интерфейс командной строки). Данные о текущем пользователе программа будет получать из системы и её файлов конфигурации. Обмен сообщениями реализован по средствам синхронизации содержимого файлов между пользователями и занесения их в буффер программы. Для каждого этапа общения (Написание, отправка, принятие, вывод) программа не будет останавливаться благодаря двухпоточной работе.

2. Обзор системы

2.1 Обзор вариантов использования

Так как приложение будет распространяться как в виде терминальной программы, так и полноценного мессенджера с GUI, существует необходимость создания схожего функционала для обоих версий. Так как программа предоставляется в качестве иходного кода и скриптов запуска с лицензией “GNU GPL 3.0”, пользователи свободны редактировать и изменять как программу, так и функционал под свои условия и нужды.

2.2 Предположения и зависимости

Предполагается необходимость использования небольшого количества инструментов и библиотек, не входящих в стандартный набор утилит Python. На данном этапе список этих зависимостей выглядит следующим образом:

PyQt5, pickle, pyperclip.

3. Детальные требования

3.1 Описание вариантов использования

Пользователь может:

Войти в систему, подключиться к собеседнику, написать сообщение, отправить сообщение, принять сообщение.

В терминальной версии (CLI) выполнение дополнительных команд происходит при их вводе в строку запроса. В версии с GUI (Qt) эти команды выполняются через отведённые виджеты.

3.2 Дополнительные требования:

ОС:

Для первой версии необходимо иметь ОС на базе ядра Linux (Тестирование и написание программы проводится на Arch Linux, но особенности разработки приложений для Linux позволяют использовать интерпретируемые программы в любых дистрибутивах). В дальнейшем возможен перенос на Windows.

Shell (для версии TUI):

Тестирование проводится в оболочке fish, но программа может работать в любой другой (bash, zsh, etc).

Qt (для версии GUI):

QT 5.12.2, стандартный набор инструментов Qt, доступный в репозитории вашего дистрибутива (Arch Linux: qt5-base, qt5-tools).

Python:

В процессе разработки использовался Python версии 3.9.7.

3.3 Дополнительные функциональные требования

Библиотеки Python:

PyQt5, pickle, sys, os, time, threading.

3.4 Другие функциональные требования

Так как разработка GUI версии ведётся на Qt, то вероятны проблемы при запуске на WayLand-сессии графического сервера. Рекомендуется использовать стандартный X-сервер.

Для тайлинговых оконных менеджеров (i3WM, DWM, awesomeWM, swayWM и других) интерфейс программы будет растягиваться на заданную область. Для корректного отображения расположения виджетов рекомендуется добавить Qt приложениям модификатор floating или переводить окно в floating-mode.

Если вы используете темы оформления Qt, то для их корректной работы в конфигурационном файле вам необходимо прописать вашу тему. Стандартная тема оформления мессенджера – breeze.