**Техническое задание и описание проекта «CryptoDog»**

**Цель.** Создать приложение с графическим интерфейсом, написанное на алгоритмическом языке высокого уровня Python с использованием фреймворка PyQt5.

**Функциональность:**

* Работа с кодировками (Base64)
* Работа с алгоритмами хэширования (MD5, SHA1, SHA256, SHA512)
* Работа с симметричными шифрами (AES, DES, 3DES, RC4)
  + - * + Возможность сохранения пользовательских ключей шифрования
* Менеджер ключей для Gpg4win:
  + Управление сертификатами OpenPGP (создание, удаление, импорт, экспорт и др.)
  + Шифрование и дешифрование данных асимметричными алгоритмами
  + Подпись и верификация данных

**Описание.** CryptoDog – графический менеджер сертификатов OpenPGP на основе утилиты Gpg4win и криптографический инструмент шифрования данных. CryptoDog предоставляет набор функционала в области криптографии и шифрования необходимый среднестатистическому пользователю.

**Требования к проекту:**

* Отзывчивость и интуитивность интерфейса
* Безопасность шифрования данных
* Безопасность хранения ключей сертификатов OpenPGP

**Используемые технологии:**

* Фреймворк *PyQt5* для реализации графического интерфейса
* Встроенная библиотека *sqlite* для работы с базой данных
* Встроенная библиотека *datetime* для работы с объектами даты и времени

Для реализации вышеописанного функционала самостоятельно были изучены следующие библиотеки:

* + Встроенная библиотека *hashlib* для работы с алгоритмами хэширования
  + Встроенная библиотека *base64* для работы с кодировкой Base64
  + Встроенная библиотека *binascii* для обработки исключений
  + Встроенная библиотека *webbrowser* для открытия ссылок
  + Библиотека *pycryptodome* для работы с симметричным шифрованием
  + Библиотека *python-gnupg* для работы с GPG
  + Библиотека *pyperclip* для работы с буфером обмена

При создании графического интерфейса также были использованы асcеты темы KDE Breeze

**Используемые классы:**

|  |  |  |
| --- | --- | --- |
| Файл | Класс | Описание |
| main.py | MainForm | Форма главного окна, его связь с остальным кодом и работа менеджера GPG |
| AboutForm | Форма окна About |
| cipher.py | Hash | Содержит в себе четыре экземпляра для работы с алгоритмами хэширования |
| B64 | Содержит в себе два экземпляра для шифрования и дешифрования шифром Base64 |
| AESx | Инициализатор принимает пользовательский ключ шифрования и длину ключа в битах (по умолчанию 128). Содержит в себе два экземпляра для шифрования и дешифрования шифром AES |
| SingleDES | Инициализатор принимает пользовательский ключ. Содержит в себе два экземпляра для шифрования и дешифрования шифром DES |
| TripleDES | Инициализатор принимает пользовательский ключ. Содержит в себе два экземпляра для шифрования и дешифрования шифром 3DES |
| RC4 | Инициализатор принимает пользовательский ключ. Содержит в себе два экземпляра для шифрования и дешифрования шифром RC4 |
| newkey\_form.py | MainForm | Форма создания новой пары ключей OpenPGP |
| sym\_passwords\_form.py | MainForm | Форма выбора сохранённого пароля для симметричного шифрования |
| notepad.py | MainForm | Форма для шифрования обычного текста ключами OpenPGP |

**Примечание.** Стабильная работа приложения гарантируется только **Python 3.11**! Так же для работы приложения на компьютере должна быть установлена утилита Gpg4win.