Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Системное программное обеспечение вычислительных машин

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему

«Email-клиент»

БГУИР КП 1-40 02 01 16\* ПЗ

Студент: гр. 550502 Кузьменко А.А.

Руководитель: ассистент кафедры ЭВМ

Лавникевич Д.А.

Минск, 2017

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

УТВЕРЖДАЮ

Заведующий кафедрой

*––––––––––––––––––––––––*

(подпись)

––––––––––––––––––––––––2017 г.

ЗАДАНИЕ

по курсовому проектированию

Студенту    *Кузьменко Алексею Андреевичу–––––– –––––‑–––*

1. Тема проекта   Email-клиент *––––––––– ––– ––‑–––*

2. Срок сдачи студентом законченного проекта –––––  *8 июня 2017 г. –*

3. Исходные данные к проекту *––––––––––––––––––––––*

4. Содержание расчетно-пояснительной записки (перечень вопросов, которые подлежат разработке)

*Введение. 1.Постановка задачи. 2.Моделирование предметной области и разра-ботка спецификации функциональных требований. 3.Обоснование технических– приемов. 4. Разработка алгоритмов и описание классов. 5. Руководство пользователя–– и результаты работы программы. Заключение. Литература. ––––––*

5. Перечень графического материала (с точным обозначением обязательных чертежей и графиков)

*1. Схема алгоритма getNumberOfEmails()*

*2. Схема алгоритма login (QString host, int port, QString user, QString password)*

6. Консультант по проекту (с обозначением разделов проекта) *Лавникевич Д.А.*

7. Дата выдачи задания  *20 февраля 2017 г.––––––––––––––––– ––*

8. Календарный график работы над проектом на весь период проектирования (с обозначением сроков выполнения и трудоемкости отдельных этапов):

*разделы 1,2 к 05.03. – 15 %; ––––––––––––––––––––––––––––––––––– – ––*

*оформление пояснительной записки и графического материала к 08.06.2017 – 15 %*

*Защита курсового проекта с 08.06 по 15.06*

РУКОВОДИТЕЛЬ*–––––––– –– Лавникевич Д.А.*

(подпись)

Задание принял к исполнению *ББ Кузьменко А.А.*

(дата и подпись студента)

ВВЕДЕНИЕ 5

1 ПОСТАНОВКА ЗАДАЧИ 7

2 МОДЕЛИРОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ И РАЗРАБОТКА СПЕЦИФИКАЦИЙ ФУНКЦИОНАЛЬНЫХ ТРЕБОВАНИЙ 8

2.1 Обзор литературы и существующих аналогов 8

2.2 Разработка спецификации функциональных требований 10

3 ОБОСНОВАНИЕ ТЕХНИЧЕСКИХ ПРИЁМОВ ПРОГРАММИРОВАНИЯ 11

3.1 Выбор языка 11

3.2 Выбор среды для реализации Email ‑ клиента 12

4 РАЗРАБОТКА АЛГОРИТМОВ И ОПИСАНИЕ КЛАССОВ 13

4.1 Разработка алгоритмов 13

4.2 Описание классов 14

4.3 Блок схемы алгоритмов 16

5 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ И РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ 17

ЗАКЛЮЧЕНИЕ 21

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 22

ПРИЛОЖЕНИЕ А (схемы алгоритмов) 23

ПРИЛОЖЕНИЕ Б (ссылка листинга кода в GitHub) 24

**ВВЕДЕНИЕ**

Появление электронной почты можно отнести к 1965 году, когда сотрудники Массачусетского технологического института Ноэль Моррис и Том Ван Влек написали программу mail для операционной системы CTSS (Compatible Time-Sharing System), установленную на компьютере IBM 7090/7094, и, конечно же, не сразу оказало огромное влияние на окружающий мир. Однако общее развитие электронной почты с годами шло и сейчас тяжело представить человека, который не обладает своим собственным почтовым ящиком.

Электронная почта бесспорно несет в своей сущности множество плюсов – это и легковосприимчивые и запоминаемые человеком адреса, и возможность передачи как простого текста, так и форматированного, передача произвольных файлов (текстовые документы, медиафайлы, программы, архивы, и т.д.), независимость серверов, простота и надежность доставки сообщения.

Однако недостатки в электронной почте все же присутствуют. Простым примером является спам (массовые рекламные и вирусные рассылки).

Для удобной работы с электронной почтой прекрасно подходит email-клиент.

Email-клиент – это специальное программное обеспечение, которое устанавливается на компьютер пользователя с целью для хранения, получения, отправки и написания сообщений электронной почты одного или нескольких пользователей.

Для работы с почтовыми серверами используются следующие протоколы:

- POP или IMAP, через которые осуществляется прием почты.

- SMPT, с помощью которого осуществляется отправка почты.

О выборе протокола приема почты будет сказано немного позже.

Хотелось бы отметить, что в настоящее время некоторые браузеры, такие как Opera или, например, SeaMonkey имеют встроенные почтовые программы. Это объясняется тем, что для просмотра некоторых писем в формате HTML почтовые программы используют движки родственных браузеров.

**1 ПОСТАНОВКА ЗАДАЧИ**

* 1. **Задача**

Реализовать собственный Email - клиент на языке С++.

* 1. **Способ реализации задачи**

По определению Email-клиент должен реализовывать хранение, отправку, получения и запись сообщений.

Для успешной реализации Email-клиента необходимо проанализировать существующие его аналоги, подобрать список литературы и надежных Интернет-ресурсов.

В процессе реализации курсового проекта следует написать качественный и читабельный код на языке.

**2 СТРУКТУРА ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ**

**2.1 Обзор литературы и существующих аналогов**

Существует огромное множество аналогов Email-клиента, однако все же можно выявить наиболее популярную серию:

1. Microsoft Outlook:

Наиболее популярный почтовый клиент, поставляемый в составе Microsoft Office. Позволяет работать с контактами, задачами и календарем. Интегрируется с большинством корпоративных систем для совместной работы. Есть мобильные версии для смартфонов и планшетов.

1. Thunderbird:

Бесплатная почтовая программа от Mozilla, лёгкая в установке и настройке — и имеющая множество возможностей.

1. Zimbra Desktop:

Бесплатный open-source почтовый клиент для Windows, созданный на основе технологии Mozilla Prism. Содержит в своём составе, помимо почты, функции органайзера, менеджера задач, портфеля, календаря и многого другого.

1. The Bat!:

Защищенный и эффективный клиент электронной почты для Windows. Главными задачами почтовой программы The Bat! являются: сохранение конфиденциальности переписки, удобство и экономия времени при работе с почтой.

1. Windows Mail:

Обновленный Outlook Express - почтовый клиент в составе Windows 7, 8, 10. Хорошая защита от спама и фишинга. Сообщения и контакты разделены на отдельные файлы с расширением EML, это позволяет при необходимости просматривать их с помощью файлового менеджера.

1. Opera Mail:

Бесплатный, простой, легко настраиваемый почтовый клиент. Поддерживает работу с несколькими учетными записями. Встроенный спам-фильтр (с функцией обучения) показывает хорошие результаты и имеет несколько степеней защиты.

1. eM Client:

Почтовый клиент для Window. Может соединяться POP/IMAP серверами, такими как Hotmail, Yahoo, и Gmail и импортировать из других почтовых клиентов. Так же программа предлагает полнофункциональный календарь, который может даже синхронизироваться с Gmail календарём или мобильным устройством.

Это семерка лучших почтовых клиентов на данный момент.

**2.2 Разработка спецификации функциональных требований**

В качестве темы курсового проекта был выбран торрент ‑ клиент.

В ходе ознакомления и анализа известных аналогов и уже существующих реализаций торрент - клиента, был выделен ряд основных требований, которые должны выполниться при реализации программы такого рода.

Были сформулированы следующие требования к курсовому проекту:

* В соответствии с базисными функциями торрент – клиента, будет реализована закачка файлов;
* Реализация удобного, красивого и интуитивного интерфейса программы;
* Раскодирование файла с помощью Bencode кодировки и вывод данных торрент – файла;
* Работа с сокетами и QNetworkRequest запросами.

**3 ОБОСНОВАНИЕ ТЕХНИЧЕСКИХ ПРИЁМОВ ПРОГРАММИРОВАНИЯ**

**3.1 Выбор языка**

C++ - низкоуровневый язык программирования, отвечающий стандартам ANSI и Международной организации стандартов (International Standards Organization — ISO). Будучи низкоуровневым языком, C++ может генерировать весьма эффективные высокоскоростные программы.

Но это еще вовсе не все преимущества данного языка, можно предоставить целый ряд преимуществ:

1. Компилируемый язык со статической типизацией;
2. Компилятор;
3. Данный язык применим в ООП;
4. Существуют опции сборки, и существует не один компилятор;
5. Исходный код, написанный программистом, может быть

предварительно обработан и изменён (например, препроцессором).

Что касается статической типизации то именно на примере этого языка проще понять, что такое тип данных, зачем он нужен и от чего зависит.

**3.2 Выбор среды для реализации Email - клиента**

Среда разработки данного курсового проекта - Windows 10. Среда программирования - Qt Creator.

Qt Creator -  [кроссплатформенная](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) [свободная](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) [IDE](https://ru.wikipedia.org/wiki/%D0%A1%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F) для разработки на С, С++ и [QML](https://ru.wikipedia.org/wiki/QML). Разработана для работы с фреймворком [Qt](https://ru.wikipedia.org/wiki/Qt" \o "Qt). Основная задача Qt Creator — упростить разработку приложения с помощью фреймворка [Qt](https://ru.wikipedia.org/wiki/Qt" \o "Qt) на разных платформах. Поэтому среди возможностей, присущих любой среде разработки, есть и специфичные, такие как отладка приложений на QML и отображение в отладчике данных из контейнеров Qt, встроенный дизайнер интерфейсов как на QML, так и на QtWidgets.

Qt Creator предполагает переносимость на уровне исходного кода программы на несколькие операционные системы, как: Unix системы и Windows, Mac, Android, IOS и другие. Одной из самых больших плюсов данной среды является хорошая документация и наличие удобного межпроцессорного взаимодействия (сигналы и слоты), которые существенно упрощает работу с графическим интерфейсом, потоками и интернет запросами. Поддерживаемые компиляторы: [GCC](https://ru.wikipedia.org/wiki/GCC), [MinGW](https://ru.wikipedia.org/wiki/MinGW" \o "MinGW), [MSVC](https://ru.wikipedia.org/wiki/MSVC), [Linux ICC](https://ru.wikipedia.org/wiki/Intel_C%2B%2B_compiler" \o "Intel C++ compiler) и другие.

.

**4 РАЗРАБОТКА АЛГОРИТМОВ И ОПИСАНИЕ КЛАССОВ**

**4.1 MainWindow (главное окно)**

Данный класс разработан с целью установки взаимосвязи с пользователем, при чем унаследован от QMainWindow. На нем находятся основные слоты для дальнейшей работы.

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

explicit MainWindow(QWidget \*parent = 0);

~*MainWindow*();

void SetLogPass(QString pass, QString log);

private slots:

void on\_FileClose\_triggered();

void on\_IncomingMessageButton\_clicked();

void on\_WriteMessageButton\_clicked();

void on\_EnterTheMessage\_triggered();

private:

Ui::MainWindow \*ui;

QString password;

QString login;

};

**LoginWindow (авторизация пользователя)**

Данный класс необходим, чтобы приобрести данные пользователя, а именно: его адрес и пароль.

class LoginWindow : public QDialog

{

Q\_OBJECT

public:

explicit LoginWindow(QWidget \*parent = 0);

~*LoginWindow*();

private slots:

void on\_CloseButton\_clicked();

void on\_EnterButton\_clicked();

void on\_PasswordLineEdit\_textChanged();

void on\_LogonLineEdit\_textChanged();

private:

Ui::LoginWindow \*ui;

};

**SendMessageWindow ()**

Довольно обширный класс, предназначенный для отправки сообщению случайному пользователю.

class SendMessageWindow : public QDialog

{

Q\_OBJECT

public:

explicit SendMessageWindow(QWidget \*parent = 0);

~*SendMessageWindow*();

void initSMTP(const QString &user, const QString &pass, const QString &host, int port = 465, int timeout = 30000);

void sendMail( const QString &from, const QString &to, const QString &subject, const QString &body);

void DeleteSMTP();

void initLogPas(QString pas, QString log);

signals:

void status( const QString &);

private slots:

void on\_SendButton\_clicked();

void stateChanged(QAbstractSocket::SocketState socketState);

void errorReceived(QAbstractSocket::SocketError socketError);

void disconnected();

void connected();

void readyRead();

private:

Ui::SendMessageWindow \*ui;

int timeout;

QString message;

QTextStream \*t;

QSslSocket \*socket;

QString from;

QString rcpt;

QString response;

QString user;

QString pass;

QString host;

int port;

enum states{Tls, HandShake ,Auth,User,Pass,Rcpt,Mail,Data,Init,Body,Quit,Close};

int state;

bool SucFlag;

};

**4.2 Описание классов**

* Файл “loginwindow.h”

В данном файле находится класс, который хранит в себе методы для того, чтобы создать диалоговое окно вывода данных о пользователе: его логина и пароля.

* Файл “mainwindow.h”

В данном файле находится класс, который напрямую взаимодействует с пользователем.

* Файл “pop3.h”

В данном файле находится класс, который предназначен для приема электронной почты с сервера.

* Файл “pop3\_commands.h”

В данном файле находится enum-класс, который хранит в себе команды для работы с сокетами.

* Файл “readmail.h”

В данном файле находится класс, который предназначен для расшифровки полученного с сервера сообщения.

* Файл “sendmessagewindow.h”

В данном файле существует класс, который хранит методы для отправки письма.

**4.3 Блок схемы алгоритмов**

**4.3.1** Алгоритм функции getNumberOfEmails(). Данный метод возвращает количество входящих сообщений на пользовательском аккаунте.

Представлен в Приложении A (Чертеж 1).

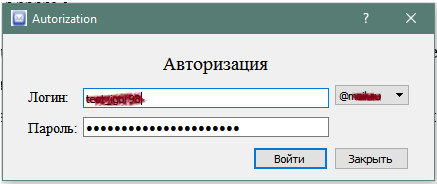
**4.3.2** Алгоритм функции login(QString host, int port, QString user, QString password). Данный метод предназначен для соединения с сервером (с помощью протокола pop3).

Представлен в Приложении A (Чертеж 2.1 и Чертеж 2.2).

**5 РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ И РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ**

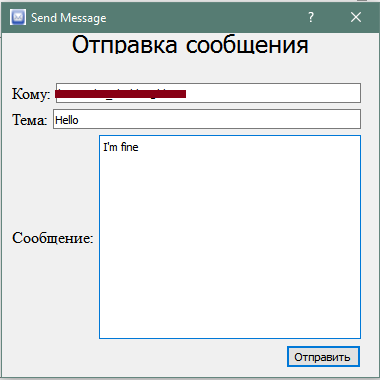
Для пользования программой необходимо произвести ее сборку и запуск в QT Creater.

Далее необходимо авторизоваться и войти в кабинет (рис 5.1):



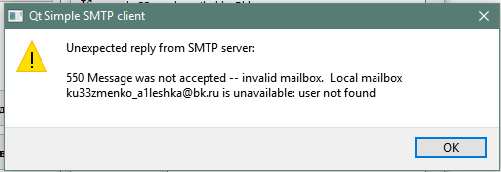
*Рис 5.1*

После успешного ввода логина и пароля необходимо выбрать дальнейший план действий, либо отправить новое сообщение случайному пользователю, либо просмотреть свои сообщения. Чтобы отправить новое сообщение необходимо нажать на кнопку «написать письмо» в левом верхнем углу (или сочетанием клавиш Ctrl+N):



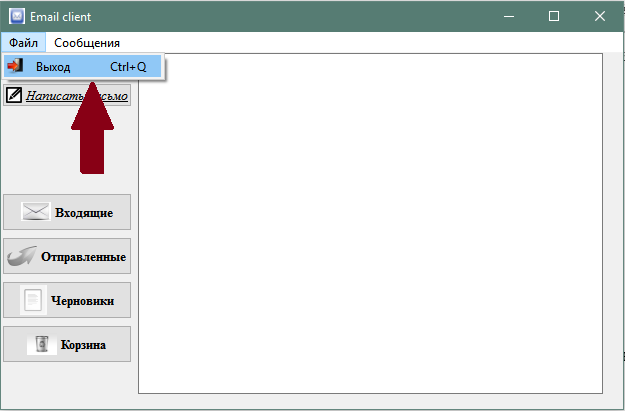
*Рис 5.2*

В случае неправильного ввода адреса пользователя, который получает письмо вылетит небольшое диалоговое окно с предупреждением и ошибкой с сервера (рис 5.3):



*Рис. 5.3*

Для завершения работы необходимо в левом углу выбрать вкладку меню и нажать «Выход» или нажать Ctrl+Q (рис 5.4):



*Рис 5.4*

**ЗАКЛЮЧЕНИЕ**

Данный курсовой проект являлся четкой реализацией поставленной задачи благодаря работы с литературой, примерами, а так же надежными интеренет-источниками.

В результате выполнения курсового проекта было разработано Email – клиент приложение, с помощью smtp и pop3 протоколов. Email – клиент позволяет пользователю отправлять, получать и хранить сообщения.

К достоинствам программы можно отнести простой, минималистичный, но не плохой инетрфейс, что в свою очередь обеспечивает удобство эксплуатации для обычных пользователей.

Данный курсовой проект продолжит свое развитие вне учебного процесса. Планируется добавить прием и передача файлов (картинок, музыки и т.д.).

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. – СПб.: Питер, 2017. – 1120с;

[2] Шлее М. - Qt 5.3. Профессиональное программирование на C++ ,2015 – 929 с;

[3] Lazar G., Penea R. - Mastering Qt 5 ,2016. – 536 с.;

[4] Страуструп, Б. Язык программирования С++ / Б. Страуструп; специаль-ное издание. Пер. с англ. – СПб. : BHV, 2008. ;

ПРИЛОЖЕНИЕ A

(схемы алгоритмов)

ПРИЛОЖЕНИЕ Б

(ссылка листинга кода в GitHub)

**ПРИЛОЖЕНИЕ Б.**

**ССЫЛКА ЛИСТИНГА КОДА В GITHUB**

Ссылка листинга кода на GitHub:

https://github.com/5ALEKSEY/Email-client