Министерство образования и науки Донецкой Народной республики

ГОУ ВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет КНТ

Кафедра ПИ

Отчет

к лабораторной работе № 2

по курсу: «Менеджмент проектов программного обеспечения»

На тему: «Разработка UML-диаграмм с использованием PlantUML»

Выполнил ст. гр.

ПИ-15а Белкин Д.Г

Проверили:

Ассистент кафедры ПИ Грищенко Д. А.

Доцент кафедры ПИ Грищенко В. И.

Донецк – 2018

**Цель лабораторной работы:** Научиться формализовать структурировать

требования к программному продукту по нечеткому словесному описанию.

Тема: Мобильное приложение «E-mail клиент»

**Описание проекта**

В качестве языка программирования был выбран Java - строго типизированный объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems.

**Сценарии использования**

**Пользователь использующий один аккаунт:** Человек, который заводит один почтовый ящик и использует только его, соответственно ему не доступна работа с несколькими аккаунтами сразу.

**Пользователь использующий два и более аккаунтов:** Данный тип пользователей может быстро переключаться между аккаунтами, а также работать с объединенным почтовым ящиком для нескольких адресов.

**Пользователь с платной подпиской:** Данному типу пользователей доступен весь функционал, а также плагин защищающий от спам-рассылок.

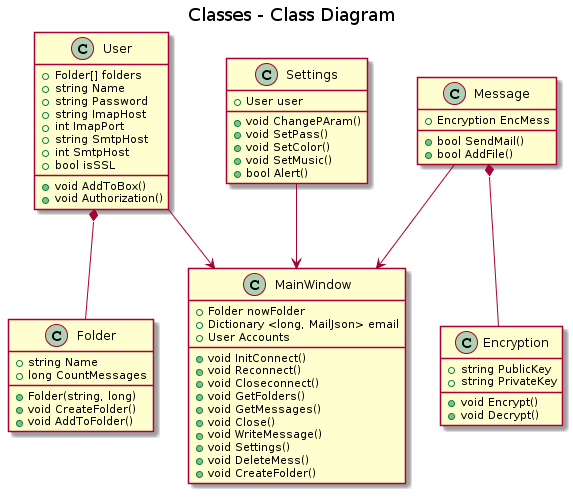


Рисунок 1 – Диаграмма классов

@startuml

title Classes - Class Diagram

class MainWindow {

+Folder nowFolder

+Dictionary <long, MailJson> email

+User Accounts

+void InitConnect()

+void Reconnect()

+void Closeconnect()

+void GetFolders()

+void GetMessages()

+void Close()

+void WriteMessage()

+void Settings()

+void DeleteMess()

+void CreateFolder()

}

class Folder {

+string Name

+long CountMessages

+Folder(string, long)

+void CreateFolder()

+void AddToFolder()

}

class User {

+Folder[] folders

+string Name

+string Password

+string ImapHost

+int ImapPort

+string SmtpHost

+int SmtpHost

+bool isSSL

+void AddToBox()

+void Authorization()

}

class Message {

+Encryption EncMess

+bool SendMail()

+bool AddFile()

}

class Settings {

+User user

+void ChangePAram()

+void SetPass()

+void SetColor()

+void SetMusic()

+bool Alert()

}

class Encryption {

+string PublicKey

+string PrivateKey

+void Encrypt()

+void Decrypt()

}

Message \*-- Encryption

Message --> MainWindow

Settings --> MainWindow

User --> MainWindow

User \*-- Folder

@enduml

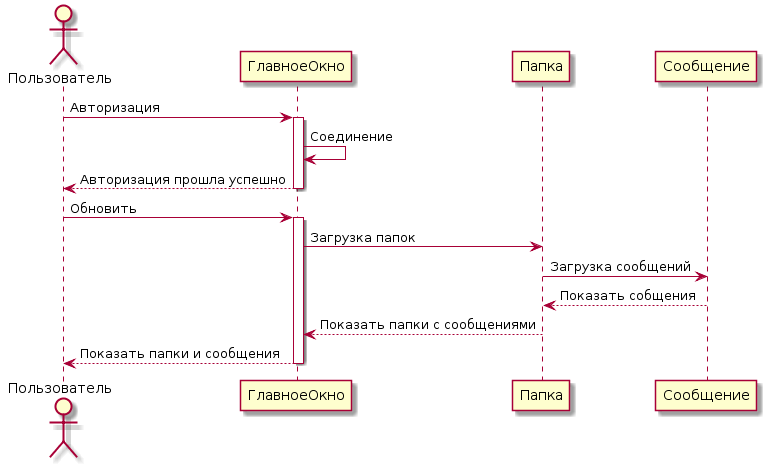


Рисунок 2 – Диаграмма последовательности для пользователя, использующего один аккаунт

@startuml

actor Пользователь

Пользователь -> ГлавноеОкно: Авторизация

activate ГлавноеОкно

ГлавноеОкно -> ГлавноеОкно: Соединение

ГлавноеОкно --> Пользователь: Авторизация прошла успешно

deactivate ГлавноеОкно

Пользователь -> ГлавноеОкно: Обновить

activate ГлавноеОкно

ГлавноеОкно -> Папка: Загрузка папок

Папка -> Сообщение: Загрузка сообщений

Сообщение --> Папка: Показать собщения

Папка --> ГлавноеОкно: Показать папки с сообщениями

ГлавноеОкно --> Пользователь: Показать папки и сообщения

deactivate ГлавноеОкно

@enduml

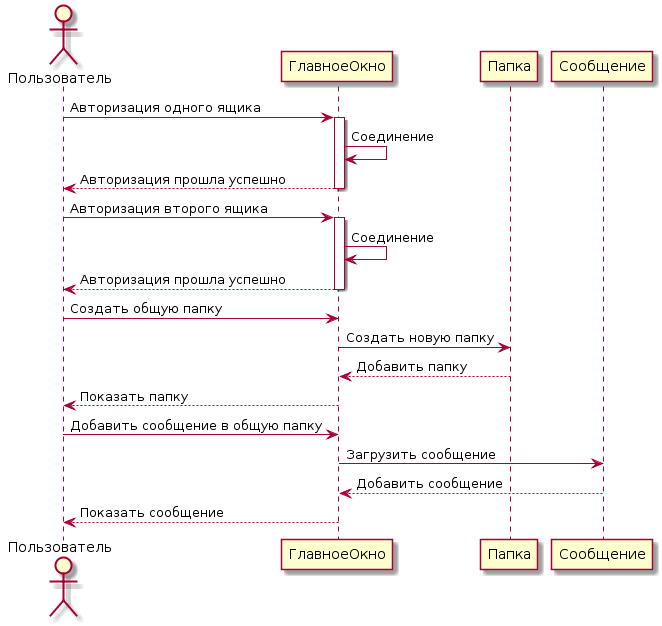


Рисунок 3 - Диаграмма последовательности для пользователя, использующего несколько аккаунтов

@startuml

actor Пользователь

Пользователь -> ГлавноеОкно: Авторизация одного ящика

activate ГлавноеОкно

ГлавноеОкно -> ГлавноеОкно: Соединение

ГлавноеОкно --> Пользователь: Авторизация прошла успешно

deactivate ГлавноеОкно

Пользователь -> ГлавноеОкно: Авторизация второго ящика

activate ГлавноеОкно

ГлавноеОкно -> ГлавноеОкно: Соединение

ГлавноеОкно --> Пользователь: Авторизация прошла успешно

deactivate ГлавноеОкно

Пользователь -> ГлавноеОкно: Создать общую папку

ГлавноеОкно -> Папка: Создать новую папку

Папка --> ГлавноеОкно: Добавить папку

ГлавноеОкно --> Пользователь: Показать папку

Пользователь -> ГлавноеОкно: Добавить сообщение в общую папку

ГлавноеОкно -> Сообщение: Загрузить сообщение

Сообщение --> ГлавноеОкно: Добавить сообщение

ГлавноеОкно --> Пользователь: Показать сообщение

@enduml

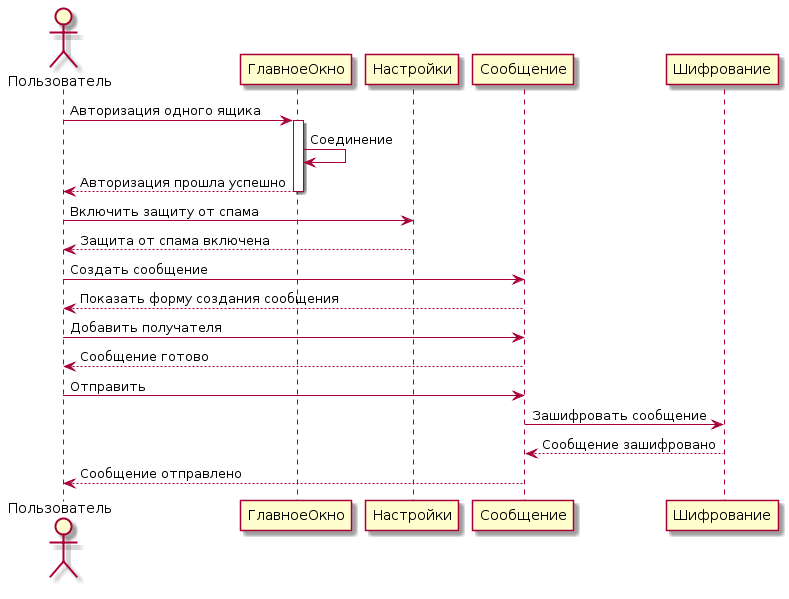


Рисунок 3 - Диаграмма последовательности для пользователя с премиум подпиской

@startuml

actor Пользователь

Пользователь -> ГлавноеОкно: Авторизация одного ящика

activate ГлавноеОкно

ГлавноеОкно -> ГлавноеОкно: Соединение

ГлавноеОкно --> Пользователь: Авторизация прошла успешно

deactivate ГлавноеОкно

Пользователь -> Настройки: Включить защиту от спама

Настройки --> Пользователь: Защита от спама включена

Пользователь -> Сообщение: Создать сообщение

Сообщение --> Пользователь: Показать форму создания сообщения

Пользователь -> Сообщение: Добавить получателя

Сообщение --> Пользователь: Сообщение готово

Пользователь -> Сообщение: Отправить

Сообщение -> Шифрование: Зашифровать сообщение

Шифрование --> Сообщение: Сообщение зашифровано

Сообщение --> Пользователь: Сообщение отправлено

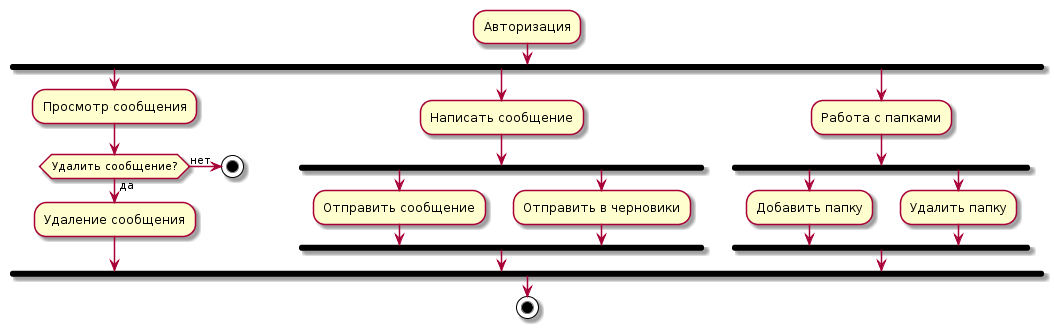
@enduml

Рисунок 4 - Диаграмма деятельности

@startuml

:Авторизация;

fork

:Просмотр сообщения;

if (Удалить сообщение?) then (да)

:Удаление сообщения;

else (нет)

stop

endif

fork again

:Написать сообщение;

fork

:Отправить сообщение;

fork again

:Отправить в черновики;

endfork

fork again

:Работа с папками;

fork

:Добавить папку;

fork again

:Удалить папку;

endfork

endfork

stop

@enduml