# 3 РАЗРАБОТКА

## 3.1 Реализация выбранной архитектуры в виде диаграммы пакетов

На основании выбранной архитектуры можно разработать диаграмму пакетов, которая представлена ниже на рисунке 3.1.

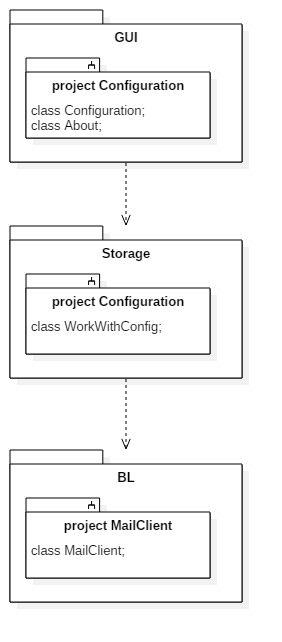


Рисунок 3.1 – Диаграмма пакетов выбранной проекта

GUI, в роли которого выступает проект Configuration, обращается к Storage и с помощью методов класса WorkWithConfig сохраняет настройки для работы службы. Как следует из рисунка GUI напрямую не взаимодействует с BL, это происходит через прослойку Storage, таким образом, пользователь может изменять настройки работы службы в удобном интерфейсе, и для этого ему не нужно знать язык XML.

## 3.2 Разработка диаграммы классов

На рисунке 3.2 приведена общая диаграмма классов, без проекции на архитектуру проекта, в следующем разделе будет приведена она же, разбита на пакеты, которая даст полное представление о структуре проекта.

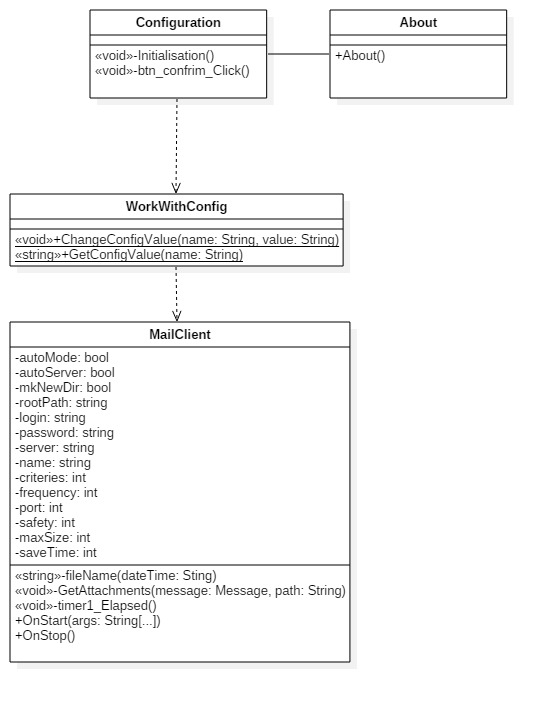


Рисунок 3.2 – Диаграмма классов проекта

Исходя из задания проекта, сложно выделить определённую сущность от которой можно было бы создать множество объектов, поэтому сущности были выделены из функций, за которые они отвечают.

В итоге были выделены классы, описание которых представлено в таблице 3.1.

Таблица 3.1 – Описание классов данного проекта

|  |  |
| --- | --- |
| **Название класса** | Описание класса |
| Configuration | Функцией данного класса является формирование GUI и связь с пакетом Storage. Класс имеет множество обработчиков событий для формирования GUI. |
| WorkWithConfig | Данный класс реализует функции работы с хранилищем, в роли которого в данном проекте выступает файловая система, а точнее конфигурационный файл расположенный в ней. Класс имеет методы*,* с помощью которых можно открыть конфигурационный файл; проверить его корректность; создать конф. файл с набором значений по умолчанию, если конфигурационный файл не был найден; прочитать и установить определённую настройку из /в конфигурационный файл соответственно. |
| MailClient | Данный класс реализует всю бизнес логику проекта. Он реализует службу, которая общается с сервером, получает сообщения, достаёт из них прикреплённые файлы и т.д. |

## 3.3 Общая диаграмма классов с отображением на архитектуру

На приведенной ниже гибридной диаграмме пакетов и классов показаны основные связи. Связи представлены на различных уровнях, как между пакетами, так и между классами. Связи между пакетами показаны с помощью методов, которые ссылаются на классы других пакетов, тем самым осуществляя взаимосвязь между отдельными частями архитектуры.

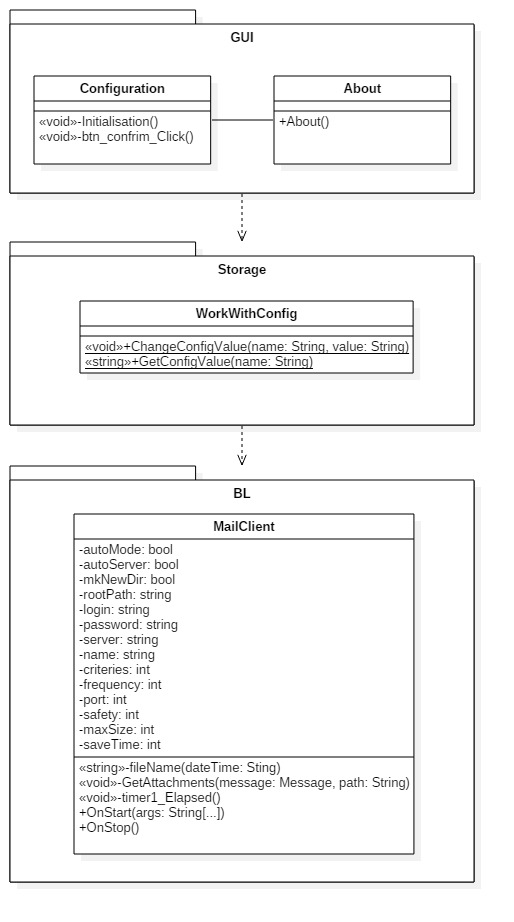


Рисунок 3.3 – Диаграмма классов проекта с проекцией на архитектуру

Благодаря рисунку 3.3 можно понять общую структуру разрабатываемого проекта. Как и описывалось ранее пользователь с помощью графического интерфейса устанавливает все настройки для службы в конфигурационный файл. Затем служба обращается к этому файлу и в соответствии заданными настройками выполняет свои функции.