

软件课设：邮件管理系统

一 开发环境

操作系统	解释器	依赖模块	图形开发组件
------	-----	------	--------

Windows	python>=3.4.4	pyqt5	QtDesigner
---------	---------------	-------	------------

为获得最佳用户体验，请使用 **Windows8**，**python3.4.4** 运行并测试

脚本执行方法

在Src目录下，运行 `python3 pxmail.py`。

可执行文件运行方法

直接运行Bin目录下已打包好的可执行文件（pxmail.exe）

二 基本特点

1. 采用python3编写，编码风格基本符合Google Style Guider。
2. 结构清晰，代码耦合度低，可扩展性强，其中：正则引擎，GUI，邮件解析器等，皆为通用组件。
3. 源代码注释详尽
4. 实现效率高，核心模块方法都是线性级别。
5. 跨平台，可在Linux、Windows、OSX运行

三 实现功能

基本功能

题目要求的基本功能全部实现：

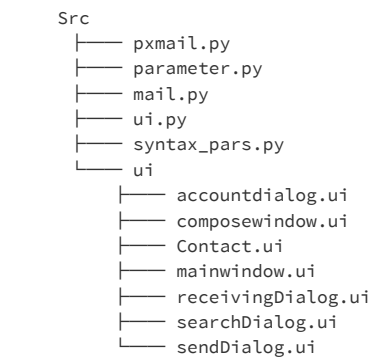
1. 有较良好的图形界面
2. 支持邮件的收发
3. 支持邮件的转发、回复
4. 支持邮件管理：浏览、查找、删除
5. 支持QQ、sina、163、hust邮箱
6. 支持html以及文本邮件的接收显示

拓展功能

1. 支持带附件邮件接收和发送
2. 支持邮件发信人、主题、内容检索
3. 按日期、发信人、主题邮件排序，支持升序、降序
4. 邮件搜索采用正则表达引擎
5. 实现了邮件内容语法高亮，提高用户体验

四 项目架构

1. 源码目录结构



简单说明：

- pxmail.py 入口主程序
- parameter.py 包含了一些定义的全局变量
- mail.py 邮件后端接收以及解析
- ui/ 界面的实现ui文件

2. 程序结构

- ui.py
GUI主要接口，实现了登陆、浏览、发送等界面

```
class AccountDialog(QtWidgets.QMainWindow) #登陆界面
class MainWindow(QtWidgets.QMainWindow): #邮件浏览主界面
class ComposeWindow(QtWidgets.QMainWindow) #发送邮件界面
class ReceiveDialog(QtWidgets.QDialog) #接收邮件对话框
class SendDialog(QtWidgets.QDialog) #发送邮件对话框
```

主界面MainWindow

```
重要接口：
def OnActivated() #排序选项Combobox
def mailDisplay() #邮件按排序显示目录
def save_binary_file() #附件保存为二进制文件
def openFile() #打开附件
def onComposeMail() #打开发送邮件界面
def onRefresh() #刷新邮件列表
def onContactList() #显示联系人列表
def InitSearchEdit() #初始化搜索框
def onReply() #回复邮件
def onForward() #转发邮件
def onDelete() #删除邮件
def onFolderSelected() #显示邮件夹目录
def onMailSelected() #显示邮件内容
```

登陆界面AccountDialog

```
重要接口：
def hideManualSet() #隐藏手动设置
def showManualSet() #展开手动设置
def txtuserEdited() #文本编辑完成自动显示服务器文本
def ontextChanged() #文本框自动补全
def onLogin() #登陆邮箱
```

发送界面ComposeWindow

重要接口：

```
def onAttachment()          #添加附件
def ontextChanged()         #文本框自动补全
def onSend()                #发送邮件
```

- mail.py
包含了邮件接收、发送线程，邮件解析器，邮件缓存机制

```
class loadingThread(QQtCore.QThread)    #登陆线程
class sendingThread(QQtCore.QThread)    #邮件发送线程
class receiveThread(QQtCore.QThread)    #邮件接收线程
class MailCache()                       #邮件缓存、记录时间戳
def get_info(msg, indent = 0)           #邮件解码，获取内容
```

邮件缓存MailCache

重要接口：

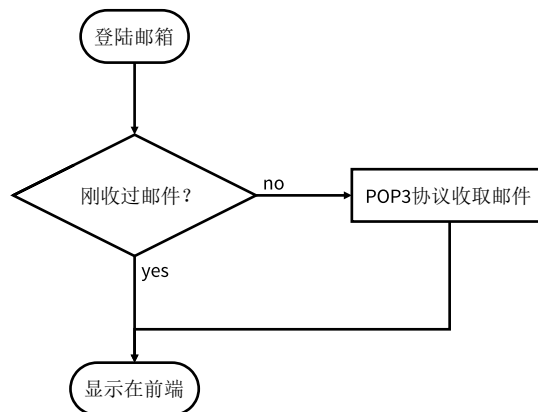
```
def _is_stale(self, folder)            #检测上一次时间戳
def _renew_state(self, folder)         #刷新时间戳
def _load_state(self)                  #读取时间戳
def _commit_state(self)                #写入时间戳
```

- syntax_pars.py
搜索引擎语法高亮机制

```
def format(color, style='')            #返回给定属性的格式
class PythonHighlighter(QSyntaxHighlighter) #对Python语言的语法高亮
```

五 程序逻辑

接收流程



1. 先从登陆界面进入主界面
2. 调用 onFolderSelected(self, folder) 选择当前显示目录
3. 启动 receiveThread() 接收线程
4. 调用 MailCache() 创建当前用户的缓存邮件夹，记下当前时间戳
5. 若当前时间遇上一次时间戳在一个周期内，则直接显示在前端，否则就先采用POP3协议从邮件服务器收取邮件到本地
6. 调用 mailDisplay(self) 输出结果

六 更新日志

2016/10/7 本次更新：

- 1.实现基本的SMTP和POP3协议收发邮件
- 2.实现了UI界面的整体框架

2016/10/23 本次更新：

- 1.解决了收取邮件时画面卡顿的问题
- 2.支持qq、sina、163、126、hust邮箱
- 3.增加了登陆失败提示
- 4.增加了回车快捷键

2016/11/3 本次更新：

- 1.增加了显示收信人、日期、主题
- 2.增加了回复、转发功能
- 3.增加了搜索功能
- 4.初步尝试对界面做了小幅改动，之后还会修改

2016/11/10 本次更新：

- 1.对邮件协议里的日期做了格式化处理
- 2.增加了对邮件按日期、收信人、主题排序
- 3.支持带附件邮件的接收，保存到本地，暂未想好如何显示到前端
- 4.搜索支持关键词背景高亮
- 5.实现了邮件内容语法高亮，提高用户体验

2016/11/12 本次更新：

- 1.增加了附件显示，能以系统默认程序打开和另存为

七 运行截图

Pxmail

用户名: phantom0506@sina.com

密码: ●●●●●●●●●●

手动设置

登录

取消

Foxmail

写邮件

刷新

通讯录

搜索邮件

常用文件夹

收件夹

草稿夹

已发送

垃圾邮件

排序: 日期

回复

转发

删除

主题:

发信人:

日期:

正在收取邮件

8%

phantom0506@sina.com

