

1 引言	2
1.1 编写目的.....	2
1.2 背景.....	2
1.3 定义.....	2
1.4 参考资料.....	3
2 总体设计.....	3
2.1 需求规定.....	3
2.2 运行环境.....	4
2.3 基本设计概念和处理流程.....	5
2.4 结构.....	5
2.5 功能需求与程序的关系.....	7
2.6 人工处理过程.....	8
2.7 尚未问决的问题.....	8
3 接口设计.....	8
3.1 用户接口.....	8
3.2 外部接口.....	8
3.3 内部接口.....	8
4 运行设计.....	9
4.1 运行模块组合.....	9
4.2 运行控制.....	9
4.3 运行时间.....	9
5 系统数据结构设计.....	9
5.1 逻辑结构设计要点.....	9
5.2 物理结构设计要点.....	10
5.3 数据结构与程序的关系.....	10
6 系统出错处理设计.....	10
6.1 出错信息.....	10
6.2 补救措施.....	11
6.3 系统维护设计.....	11

概要设计说明书

1 引言

1.1 编写目的

利用比较抽象的语言对目标系统的整个设计进行概括,使软件编程人员能对目标系统的设计细节有一致的认识,确定对系统的物理配置,确定整个系统的处理流程和系统的数据结构,接口设计,人机界面,实现对系统的初步设计。建立起目标系统的逻辑模型。

本阶段完成系统的大致设计并明确系统的数据结构与软件结构。本设计说明书的目的就是进一步细化软件设计阶段得出的软件概貌,把它加工成在程序细节上非常接近于源程序开发的软件表示。在软件项目的开发过程中起到引导作用,保证项目小组按时保质地完成项目目标,便于项目团队成员更好地了解项目情况,使项目的实际开发过程合理有序。因此以文档化的形式,记录下软件的总体设计、接口设计、运行设计、数据结构设计、系统出错处理设计,作为开发过程中项目团队成员之间的共识与约定,以及项目团队开展和检查项目工作的依据。

本设计说明书仅供开发小组成员参考。

1.2 背景

本项目全称为《Clean Mailbox》,由小组成员张森、严灏共同提出设计,并由张森、严灏协同开发的一款基于机器学习算法的垃圾邮件过滤系统。本软件需要在 windows XP 及以上系统,连接互联网的环境下运行,并可以完成在互联网环境下收发邮件、标记删除垃圾邮件等功能。

本软件需要通过邮箱账户获取来自电子邮件服务商的账户数据,并通过对服务商发送请求完成电子邮件的相关操作。

1.3 定义

- **系统:** 若未特别指明,统指本邮箱软件;
- **QT Creator:** Qt Creator 是跨平台的 Qt IDE,Qt Creator 是 Qt 被 Nokia 收购后推出的一款新的轻量级集成开发环境(IDE)。此 IDE 能够跨平台运行,支持的系统包括 Linux (32 位及 64 位)、Mac OS X 以及 Windows;
- **Anaconda:** 一个开源的 Python 发行版本,包含了 conda、Python 等 180 多个科学包及其依赖项;
- **分类器:** 在已有数据的基础上学习的一个分类函数或构造出的一个分类模型;
- **SQA:** 软件质量保证(SQA-Software Quality Assurance)是建立一套有计划,有系统的方法,来向管理层保证拟定出的标准、步骤、实践和方法能够正确地被所有项目所采用。

1.4 参考资料

《项目开发计划 2.0》见数据仓库 <https://github.com/airzs/EmailBox/>

《软件需求规格说明书 2.0》见数据仓库 <https://github.com/airzs/EmailBox/>

《软件工程——实践者的研究方法》 Roger S. Pressman, Bruce R. Maxim 编著，机械工业出版社，2015

《数据库系统概念》 Abraham Silberschatz, Hennry F. Korth, S. Sudarshan 编著，机械工业出版社，2012

《统计学习方法》 李航编著，清华大学出版社，2012

《机器学习》 周志华编著，清华大学出版社，2016

2 总体设计

2.1 需求规定

本项目主要的输入输出项目有写信、发送邮件、回复邮件等等，下表列出了各个功能的名称、简述、输入、输出以及处理算法。

编号	名称	简述	输入	处理(算法)	输出
1	登录	用户输入账号密码登录邮箱	账号、密码	验证账号密码正确性并返回结果	若账号密码正确，进入邮箱主界面，否则提示用户输入错误
2	保存账号密码	对用户账号密码进行保存	选择“保存账号密码”功能	对所输入账号密码进行储存	在登录界面显示已保存账号信息
3	写信	用户进行邮件输入	收件人、主题、正文	对所输入内容进行储存	保存用户输入内容并在界面显示所有内容
4	发送邮件	用户通过点击发送进行邮件发送	点击“发送”功能	对邮件进行数据结构转换并通过协议发送邮件	若发送成功，提示用户操作成功，否则提示用户发送失败

5	查看邮件	用户通过收件箱查看相应邮件	点击收件箱并选择相应邮件	根据用户选择提取相应的邮件并显示全部内容	显示用户所选择邮件的所有内容
6	回复邮件	用户对当前查看邮件进行回复	点击“回复”功能	提取当前邮件信息并进入写信界面，提示用户输入回复内容	进入写信界面并且已标识收件人
7	删除邮件	用户可选择删除无用邮件	选中相应邮件并选择“删除”功能	对用户所选取的邮件进行删除并删除其储存	刷新邮箱并提示用户成功删除邮件
8	标识垃圾邮件	用户选择垃圾邮件并标识移入垃圾箱	选中相应邮件并选择“移入垃圾箱”功能	将用户所选取的邮件移入垃圾箱	将邮件从收件箱移入垃圾箱并刷新界面
9	邮件存草稿	用户将当前输入邮件进行存草稿	点击“存草稿”功能	对用户当前操作邮件进行存草稿移入草稿箱	在草稿箱生成当前邮件副本
10	垃圾邮件过滤	对已收邮件进行垃圾邮件过滤，将垃圾邮件移入垃圾箱	所有邮件内容	通过学习算法对邮件中出现的关键词进行解析过滤，标识垃圾邮件并进行移动处理	将垃圾邮件移入垃圾箱，普通邮件放入收件箱
11	退出邮箱	用户选择安全退出邮箱	点击“退出”功能	根据用户点击退出邮箱系统，返回登录主界面	返回登录主界面

2.2 运行环境

2.2.1 设备

- Windows PC 一台；
- 酷睿 i3 以上的处理芯片一枚及以上；
- 1GB 显存及以上的兼容显卡；

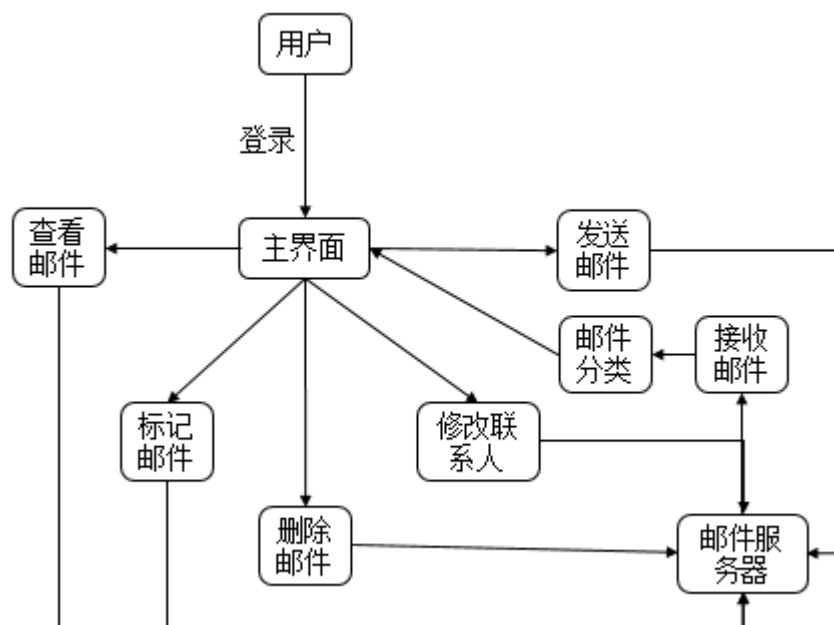
- 500M 的存储空间及以上
- Internet 互联网接入
- 联机 VGA 彩显一台

2.2.2 支持软件

- 操作系统：Windows XP 及以上操作系统
- 编程开发软件：Visual Studio 2015，QT Creator 5.7.0，Anaconda 3.4.0

2.3 基本设计概念和处理流程

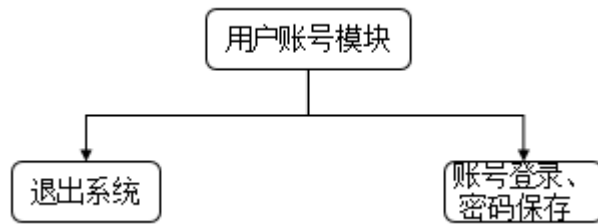
本系统的处理流程如下图所示：



2.4 结构

用一览表及框图的形式说明本系统的系统元素（各层模块、子程序、公用程序等）的划分，扼要说明每个系统元素的标识符和功能，分层次地给出各元素之间的控制与被控制关系。

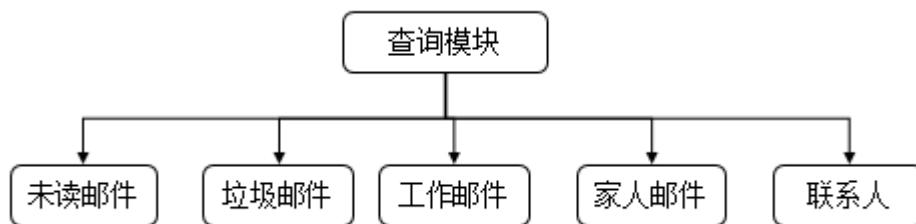
（1）用户帐号模块：



用户帐号模块包括：用户设置，退出系统二个部分。

1. 用户设置：实现系统用户登录和密码保存。
2. 退出系统：实现正常退出邮箱系统。

(2) 查询模块：



查询模块包括未读、垃圾、工作、家人、联系人五个部分。

1. 查询未读邮件：查询用户未读的邮件；
2. 查询垃圾邮件：查询被分类为或标记为垃圾的邮件；
3. 查询工作邮件：查询被分类为或标记为工作的邮件；
4. 查询家人邮件：查询被分类为或标记为家人的邮件；
5. 查询联系人：用户的联系人；

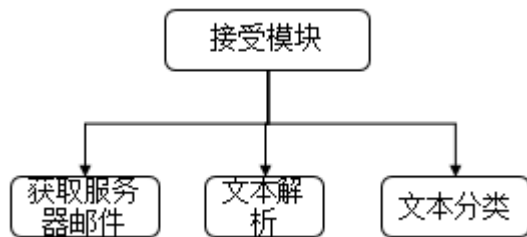
(3) 写信模块：



写信模块包括：邮件编辑，草稿箱，邮件发送三个部分。

1. 邮件编辑：实现用户对邮件的编辑。
2. 草稿箱：用户将未编辑完的邮件存于草稿箱内。
3. 发送邮件：将编辑好的邮件发送至邮件服务商的服务器上

(4) 接受邮件模块：



接受模块包括：获取服务器邮件，文本解析，文本分类三个部分。

1. 获取服务器邮件：从邮件服务提供商服务器上得到邮件。
2. 文本解析：将得到的邮件进行文本解析。
3. 文本分类：将解析好的数据输入到分类器进行文本分类。

(5) 信息修改模块：



修改信息模块包括：标记邮件，删除邮件，修改联系人信息三个部分。

1. 标记邮件：用户对邮件进行标记服务。
2. 删除邮件：用户删除无用邮件。
3. 修改联系人信息：用户添加、删除、修改联系人信息。

2.5 功能需求与程序的关系

本条用一张如下的矩阵图说明各项功能需求的实现同各块程序的分配关系：

	模块 1	模块 2	模块 3	模块 4	模块 5
登录	√				
保存账号密码	√				
写信			√		
发送邮件			√		
查看邮件		√			
标识邮件是否已读					√
回复邮件			√		
删除邮件					√
标识垃圾邮件					√
邮件存草稿			√		

通讯录					√
垃圾邮件过滤				√	
退出邮箱	√				
邮件存草稿			√		

2.6 人工处理过程

需要手动输入数据，及窗口或菜单栏选项进行信息确认或者选择，需要用户对邮件进行标记以帮助分类器对邮件进行分类。

2.7 尚未问决的问题

服务器数据库数据与本机数据库的不同步问题尚待解决。

3 接口设计

3.1 用户接口

用户接口采用窗口式、按钮以及菜单栏进行设计，用户直接对用户界面进行点击便可以进行本系统的一系列操作，操作简单，不需要任何命令与语法结构，减少用户记忆负担，更便于用户使用。

3.2 外部接口

硬件接口：无硬件接口；

服务器接口：系统与 POP、SMTP 邮箱服务器之间的通信接口。

数据库的接口：QT Designer 与数据库之间的接口

3.3 内部接口

通过面向对象语言设计类，在 public 类中实现调用；类间实现严格封装。各个模块之间采用函数调用、参数传递、返回值的方式进行信息传递。

4 运行设计

4.1 运行模块组合

本程序主要是以一个窗口为模块，系统有一个主窗口，可以在主窗口上完成大多数操作，某些操作需要主窗口通过打开另一个子窗口来实现模块之间不同功能的连接和组合。各模块之间相对独立，程序的可移植性好。各模块之间主要以传递数据项的引用来实现模块之间的合作和数据共享。

4.2 运行控制

用户接口采用窗口式、按钮以及菜单栏进行设计，用户直接对用户界面进行点击便可以进行本系统的一系列控制与操作。

4.3 运行时间

各模块运行时间不定，这也跟用户的操作以及数据的大小有关，确保上机流畅。

5 系统数据结构设计

5.1 逻辑结构设计要点

给出本系统内所使用的每个数据结构的名称、标识符以及它们之中每个数据项、记录、文卷和系的标识、定义、长度及它们之间的层次的或表格的相互关系。

用户信息数据字典：

属性名	存储代码	类型	备注
标识符	Id	char	用户编号
邮箱账号	Name	char	用户账号
邮箱密码	Pswd	char	用户密码

邮件信息数据字典：

属性名	存储代码	类型	备注
标识符	ID	int	邮件唯一标识符
发送者	Sender	char	邮件发送者
接受者	Receiver	char	邮件接收者

日期	Time	char	发送时间
标题	Title	Char	邮件标题
正文	Body	Char	邮件正文
类别	Variety	Int	邮件类别
有效位	Valid	Int	邮件有效位

草稿信息数据字典：

属性名	存储代码	类型	备注
标识符	ID	int	草稿唯一标识符
发送者	Sender	char	邮件发送者
接受者	Receiver	char	邮件接收者
标题	Title	Char	邮件标题
正文	Body	Char	邮件正文

5.2 物理结构设计要点

数据的物理操作有数据库解决，不在本系统开发的考虑范围之内。

5.3 数据结构与程序的关系

数据结构与程序是软件的重要组成部分，程序的正确执行依赖于合理的数据结构

6 系统出错处理设计

6.1 出错信息

故障类型	故障原因	系统输出
非法输入	用户输入信息不合法	输入信息错误，请重新输入
PC 断电	用户信息丢失	程序定时保存用户工作内容， 保证用户数据不会完全丢失
网络故障	用户的网络出现问题	网络故障！
PC 端数据文件丢失	软件运行出错	数据丢失，请重新安装软件
数据库连接错误	数据库设置不正确或数据库 服务器异常	提醒用户检查数据库

6.2 补救措施

故障类型	故障后果	故障处理要求
PC 断电	用户信息丢失	程序定时保存用户工作内容， 保证用户数据不会完全丢失
网络故障	用户信息丢失	程序定时保存用户工作内容， 保证用户数据不会完全丢失
PC 端数据文件丢失	软件运行出错	尽量利用电邮服务商提供的 数据，减少本地数据依赖
数据库连接错误	数据库数据出错	提醒用户检查数据库，取消本 次操作

6.3 系统维护设计

- 在数据库设计的时候，学生信息与学生上机、学生充值、学生退卡记录都不在一个表中，防止信息删除或更新的时候，信息的不一致重叠罗嗦。
- 在程序内部通过建立模块，不同窗体调用一个模块的形式进行设计，减少了代码的操作量，提高了运行的速度。