

1 引言	1
1.1 编写目的.....	1
1.2 背景.....	1
1.3 定义.....	1
1.4 参考资料.....	2
2.1 工作内容.....	2
2.2 主要参加人员.....	2
2.3 产品.....	3
2.3.1 程序.....	3
2.3.2 文件.....	3
2.3.3 服务.....	4
2.3.4 非移交的产品.....	4
2.4 验收标准.....	4
2.5 完成项目的最迟期限.....	4
2.6 本计划的批准者和批准日期.....	5
3 实施计划	5
3.1 工作任务的分解与人员分工.....	5
3.2 接口人员.....	5
3.3 进度.....	5
3.4 预算.....	7
3.5 关键问题.....	7
4 支持条件	8
4.1 计算机系统支持.....	8
4.2 需由用户承担的工作.....	8
4.3 由外单位提供的条件.....	8
5 专题计划要点	9

项目开发计划（GB856T——88）

1 引言

1.1 编写目的

此项目开发计划书的编写主要是为了给开发《Clean MailBox》做主要的规划和整合，在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，使项目工作开展的各个过程合理有序。因此以文件化的形式，把对于在项目生命周期内的工作任务范围、各项工作的任务分解、项目团队组织结构、各团队成员的工作责任、团队内外沟通协作方式、开发进度、项目内外环境条件、风险对策等内容做出的安排以书面的方式，作为项目团队成员以及项目干系人之间的共识与约定，项目生命周期内的所有项目活动的行动基础，项目团队开展和检查项目工作的依据。

1.2 背景

本项目全称为《Clean Mailbox》，是一款基于机器学习算法的垃圾邮件过滤系统，目标用户是经常使用电子邮件，对电子邮箱内的垃圾邮件感到反感的用户。

数据显示，近年来电子邮件用户收到的垃圾邮件所占比重已经超过总邮件数的一半，且呈缓慢增长的趋势。在工作生活高度信息化的今天，大量的垃圾邮件不仅严重影响了电邮用户的体验，更加降低了用户的工作效率。如何有效的标记、删除垃圾邮件便成了一个亟待解决的问题。而我们的项目则旨在为用户更好的标记垃圾邮件，为用户删除垃圾邮件提供便利，提高用户的工作效率。

本项目由组长张森和组员严灏共同规划并提出，在 Windows 环境下运行。

1.3 定义

1.3.1 专门术语：

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；

1.3.2 缩写:

系统: 若未特别指明, 统指本邮箱软件

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所有项目所采用。

1.4 参考资料

《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

2.项目概述

2.1 工作内容

团队计划在本学期内完成此垃圾邮件过滤系统项目, 为用户提供邮件收发, 垃圾邮件标记分类等功能, 结合软件工程方法加强小组成员之间的团队协作, 做到框架的清晰、简洁。代码整体做到低聚合性和高耦合性, 重用性高, 减少代码编写过程中的错误。核心分类算法到达较理想的期望值, 前端界面用户友好, 为用户提供尽量好的使用体验。要完成以上目标, 总体来说主要需要完成以下的一些工作内容:

1. 制作和修订项目开发计划
2. 进行计划跟踪与监控
3. 配合 SQA 的质量保证
4. 工作产品及时进行受控管理
5. 按计划阶段评审
6. 小组成员测试产品
7. 完成最终工作产品
8. 项目实施总结
9. 项目验收

2.2 主要参加人员

本项目共有两名成员: 一位组长, 一位组员, 由两名成员协作完成系统设计, 代码实现, 软件测试等任务, 具体分工如下表所示:

姓名	角色	已有基础	主要任务
张森	项目领导者	具备机器学习算法知识与完整的项目开发经验	软件设计, 功能实现, 软件测试

严灏	成员	有完整的QT 项目开发经验	界面设计, 功能实现, 软件测试
----	----	---------------	------------------

2.3 产品

2.3.1 程序

a. 软件名称:

Clean MailBox--垃圾邮件过滤系统

b.提交内容:

源程序、数据库对象创建语句、可执行程序、支撑系统的数据库数据、配置文件、第三方模块、界面文件、界面原稿文件、声音文件、安装软件、安装软件源程序文件等等。

c. 编程语言:

C++, Python

d. 存储方式:

开源项目托管平台 GitHub

e. 功能:

Clean MailBox 是一个包含垃圾邮件过滤功能的简单电子邮件系统, 具备各种电子邮件应有的基本功能。用户通过本系统能登录、退出自己的邮箱, 接收、发送和删除邮件, 选择标记邮件是否为垃圾邮件, 并通过系统的提示便捷的选择删除邮箱内的大量垃圾邮件。

2.3.2 文件

使用说明: 本使用说明详细描述软件的功能、性能和用户界面, 使用户对如何使用该软件得到具体的了解, 为操作人员提供该软件各种运行情况的有关知识, 特别是操作方法的具体细节;

软件维护手册: 主要包括软件系统说明、程序模块说明、操作环境、支持软件的说明、维护过程的说明、便于软件的维护。

2.3.3 服务

无服务

2.3.4 非移交的产品

1. 项目开发计划书：为软件项目实施方案制订出具体计划，包括各部分工作的负责人员、开发的进度、所需的硬件及软件资源等。
2. 软件需求说明书：对所开发软件的功能、性能、用户 界面及运行环境等做出详细的说明。它是在用户与开发人员双方对软件需求取得共同理解并达成协议的条件下编写的，也是实施开发工作的基础。
3. 测试计划书：为做好单元测试、集成测试和验收测试，需为如何组织测试制订实施计划。
4. 使用说明：本使用说明详细描述软件的功能、性能和用户界面，使用户 对如何使用该软件得到具体的了解。
5. 测试分析报告：测试工作完成以后，应提交测试计划执行情况的说明， 对测试结果加以分析，并提出测试的结论意见。

2.4 验收标准

2.4.1 代码的验收

最后在交付项目之前进行小组内评审，确保与文档说明保持一致，代码书写风格统一，采用标准规范，没有由于软件缺陷造成丢失数据、垃圾邮件分类率过低、响应时间太长、邮件无法正常发送接收等问题，通过期末答辩的方式由助教或老师检查验收。

2.4.2 文档验收

最后在交付项目之前进行小组内评审，功能符合与小组预期的要求，文档内容逻辑清楚，清晰易读，没有语病与歧义，没有夸大或掩饰，文档整理有条理，经助教或老师检查合格后方可验收。

2.4.3 服务验收

无

2.5 完成项目的最迟期限

完成本项目的最迟期限为 2017 年 5 月 31 日

2.6 本计划的批准者和批准日期

本系统由张森，严灏提出，自 2017 年 3 月 12 日正式提交；

组员认可签字：

组长认可签字：

经理认可签字：

3 实施计划

3.1 工作任务的分解与人员分工

本项目当前成员为两人，由两人分工协作完成整个工作任务，通过建立群组保持日常联系，保证小组成员之间的交流，组长组员可通过群组共享资源。任务主要分解为五个主要的模块：软件设计、界面设计、功能实现、文档撰写和软件测试，每个模块的分工情况如下表所示：































任务分解	负责人
软件设计	张森
界面设计	严灏
功能实现	张森、严灏
文档撰写	张森、严灏
软件测试	严灏

3.2 接口人员

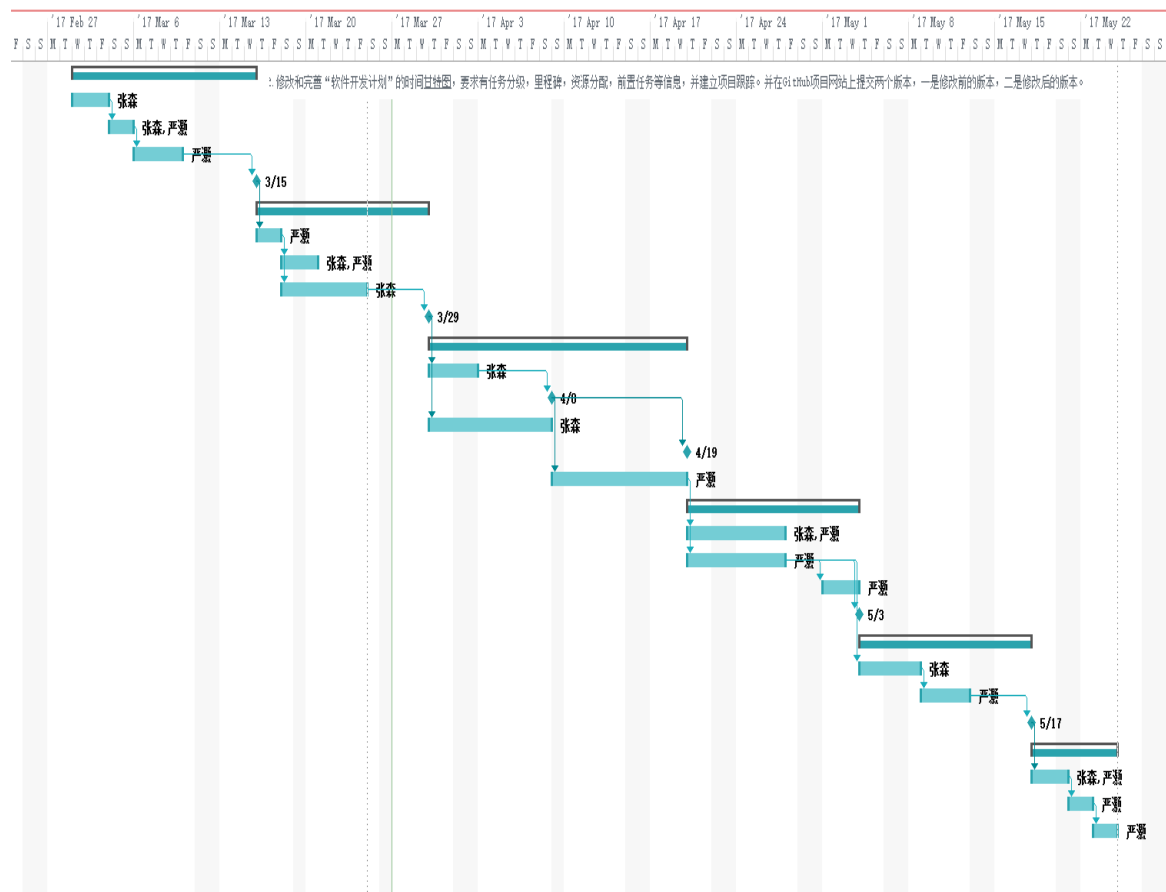
无

3.3 进度

本项目的进度安排通过甘特图的形式给出，进度表如下图所示：

		Task Mode	任务名称	Miles	Duration	Start	Finish	Predecessor	Resource Names
1			系统需求	No	13 days	Wed 17/3/1	Wed 17/3/15		张森, 严灏
2			调研用户需求	No	3 days	Wed 17/3/1	Fri 17/3/3		张森
3			讨论项目可行性	No	2 days	Sat 17/3/4	Sun 17/3/5	2	张森, 严灏
4			制定项目计划	No	4 days	Mon 17/3/6	Thu 17/3/9	3	严灏
5			规划完善项目计划书	Yes	4 days	Fri 17/3/10	Wed 17/3/15	4	张森
6			需求分析	No	12 days	Thu 17/3/16	Wed 17/3/29		张森, 严灏
7			确定系统运行环境	No	2 days	Thu 17/3/16	Fri 17/3/17	5	严灏
8			确定系统功能及性能需求	No	2 days	Sat 17/3/18	Mon 17/3/20	7	张森, 严灏
9			编写需求规格、用户说明概要	No	6 days	Sat 17/3/18	Fri 17/3/24	7	张森
10			确定项目开发计划	Yes	4 days	Sat 17/3/25	Wed 17/3/29	9	张森, 严灏
11			设计	No	16 days	Thu 17/3/30	Wed 17/4/19		张森, 严灏
12			建立总体框架结构、划分功能	No	3 days	Thu 17/3/30	Sun 17/4/2	10	张森
13			定义模块接口	Yes	6 days	Mon 17/4/3	Sat 17/4/8	12	严灏
14			数据库	No	8 days	Thu 17/3/30	Sat 17/4/8	10	张森
15			各模块的具体实现算法	Yes	9 days	Sun 17/4/9	Wed 17/4/19	13	张森, 严灏
16			制定模块测试方案	No	9 days	Sun 17/4/9	Wed 17/4/19	13	严灏
17			实现	No	10 days	Thu 17/4/20	Wed 17/5/3		张森, 严灏
18			编写程序源代码	No	6 days	Thu 17/4/20	Thu 17/4/27	16	张森, 严灏
19			模块测试和调试	No	6 days	Thu 17/4/20	Thu 17/4/27	16	严灏
20			编写用户手册	No	3 days	Mon 17/5/1	Wed 17/5/3	19	严灏
21			评审	Yes	3 days	Mon 17/5/1	Wed 17/5/3	19	张森
22			测试	No	10 days	Thu 17/5/4	Wed 17/5/17		张森, 严灏
23			单元测试	No	3 days	Thu 17/5/4	Mon 17/5/8	19	张森
24			集成测试	No	4 days	Tue 17/5/9	Fri 17/5/12	23	严灏
25			验收测试	Yes	4 days	Sat 17/5/13	Wed 17/5/17	24	张森, 严灏
26			维护	No	5 days	Thu 17/5/18	Wed 17/5/24		张森, 严灏
27			完善错误应用	No	3 days	Thu 17/5/18	Sat 17/5/20	25	张森, 严灏
28			编写故障报告以及修改报告	No	2 days	Sun 17/5/21	Mon 17/5/22	27	严灏
29			制定用户手册	No	2 days	Tue 17/5/23	Wed 17/5/24	28	严灏

甘特图如下图所示：



3.4 预算

本项目主要存在时间成本和设备成本，没有相关经费的预算，设备成本主要有：Windows 个人电脑 2 台，U 盘两个；时间预算如下表所示：

小组成员	工作量（月）
张森	3
严灏	3

3.5 关键问题

本项目的关键问题主要是一些技术上的难点和项目开发过程中一些潜在的风险可能对整个项目造成的影响。

(1). 技术难点如下表所示：

技术难点	难度	影响
文本解析	中	中
分类算法	中	中

分类器训练	小	中
邮件收发功能	中	大

(2). 风险问题如下表所示:

潜在风险	出现概率	影响	解决办法
没有足够的开发经验	中	不能及时完成项目, 或导致项目失败	请教老师, 多与有相关经验的同学交流, 在开发过程中注意总结经验
知识储备不足	中	遇到技术瓶颈, 导致整个项目开发阻塞	在项目开发过程中同步学习相关知识, 请教同学或老师
开发时间有限	中	不能及时完成项目, 影响项目答辩	每周进行一次进度报告, 确保团队进度在计划之内

4 支持条件

4.1 计算机系统支持

a. 硬件支持:

计算机: Windows PC

b. 软件支持:

开发工具: QT Creator, Visual Studio, Anaconda 等

数据库: 待定

操作系统: Windows XP 及以上的操作系统

c. 其他支持:

开发语言: Python (Anaconda), c++(Visual Studio)

账户: 可使用的能够正常访问的邮箱账户

4.2 需由用户承担的工作

无

4.3 由外单位提供的条件

无

5 专题计划要点

专题计划也就是因为项目的需要在本文档之外独立建立的计划，本节说明本项目开发中需要制定的各个专题计划的要点。本专题计划可能包括配置管理计划、质量保证计划、测试计划。

5.1 配置管理计划

编制有关软件配置管理的条款，或引用按照 GB/T 12505 单独制订《配置管理计划》文档。在这些条款或文档中，必须规定用于标识软件产品、控制和实现软件的修改、记录和报告修改实现的状态以及评审和检查配置管理工作等四方面的活动。

根据《GB/T 12505 计算机软件配置管理计划规范》，软件配置管理计划内容如下：

- 引言（包括质量计划的目的、定义、参考资料）
- 管理（描述负责软件配置管理的机构、任务、职责及其有关的接口控制。）
- 软件配置管理活动（描述配置标识、配置控制、配置状态记录与报告以及配置检查与评审等到四方面的软件配置管理活动的需求。）
- 工具、技术和方法（指明为支持特定项目的软件配置管理所使用的软件工具、技术和方法，指明它们的目的，并在开发者所有权的范围内描述其用法）

5.2 质量保证计划

执行质量评审活动，对过程质量进行控制。规模较大的项目应当单独编写《软件开发项目质量计划》。根据 GB/T 12504 计算机软件质量保证计划规范，内容包括：

- 引言（包括质量计划的目的、定义、参考资料）
- 文档（列出在该软件的开发、验证与确认以及使用与维护等阶段中需要编制的文档，并描述对文档进行评审与检查的准则）
- 评审和检查（规定所要进行的技术和管理两个方面的评审和检查工作，并进行软件需求评审、概要设计评审、软件验证与确认评审、软件系统功能检查、程序和文档物理检查）
- 软件配置管理（编制有关配置管理条款，或在“5.1 配置管理计划”中说明，或引用按照《GB/T 12505 计算机软件配置管理计划规范》单独制定的文档）
- 工具、技术和方法（指明用于支持特定软件项目质量管理工作的工具、技术和方法，指出它们的目的和用途）

5.3 测试计划

拟在 2017 年 5 月下旬对本系统进行各项测试，测试任务由小组两人协同完成，确保发现软件存在的主要问题，主要测试内容有：

1. 在 PC 上，该系统的稳定性；
2. 邮件收发等功能是否存在问题；
3. 该系统对垃圾邮件的分类是否理想。

5.4 用户培训计划

本项目组计划在邮箱系统软件完成准备交付时,由严灏同学对使用客户进行系统使用培训。