

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.....	3
2.3.4 非移交的产品.....	3
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.....	4
3 实施计划	4
3.1 工作任务的分解与人员分工.....	4
3.2 接口人员.....	5
3.3 进度.....	5
3.4 预算.....	6
3.5 关键问题.....	6
4 支持条件	7
4.1 计算机系统支持.....	7
4.2 需由用户承担的工作.....	7
4.3 由外单位提供的条件.....	7
5 专题计划要点	7

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

- a. 待开发的软件项目名称：Clean MailBox——垃圾邮件过滤系统
- b. 项目提出者：张森、严灏
项目开发者：张森、严灏
项目用户：经常使用电子邮件，对邮箱内泛滥的垃圾邮件感到反感的用户
- c. 软件系统同其他系统关系：
本软件在 Windows 环境下运行，需要使用邮件服务商的服务器。
- d. 软件系统与其他机构关系：无

1.3 定义

1.3.1 专门术语：

C++： 一门面向对象的计算机编程语言；

Python： 是一种面向对象的解释型计算机程序设计语言；

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；

GitHub： GitHub 是一个面向开源及私有软件项目的托管平台，因为只支持 Git 作为唯一的版本库格式进行托管，故名 GitHub；

分类器： 在已有数据的基础上学会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被所有项目所采用。

UML: Unified Modeling Language (统一建模语言)

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

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

姓名	角色	已有基础	主要任务
张森	项目领导者	有基本的邮件分类算法知识	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, 文档撰写, 软件测试

2.3 产品

2.3.1 程序

软件名称：Clean MailBox--垃圾邮件过滤系统

编程语言：C++, Python

存储方式：开源项目托管平台 GitHub

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，包括登录录入信息，接收、发送和删除邮件，邮件选择操作以及其他一些基础功能。

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。

2.3.3 服务

无

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。

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。

2.4.3 服务验收

服务达到文档说明的要求，小组考核合格。

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

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

进度表：

名称	开始日期	结束日期	负责人
☐ ● 系统需求	17-3-1	17-3-15	张森、严灏
● 调研用户需求	17-3-1	17-3-3	张森
● 讨论项目可行性	17-3-4	17-3-5	张森、严灏
● 制定项目计划	17-3-6	17-3-9	严灏
● 规划完善项目计划书	17-3-10	17-3-15	张森
☐ ● 需求分析	17-3-16	17-3-29	张森、严灏
● 确定系统运行环境	17-3-16	17-3-17	严灏
● 确定系统功能及性能要求	17-3-18	17-3-20	张森、严灏
● 编写需求规格、用户说明概要	17-3-18	17-3-24	张森
● 确定项目开发计划	17-3-25	17-3-29	张森、严灏
☐ ● 设计	17-3-30	17-4-19	张森、严灏
● 建立总体框架结构、划分功能	17-3-30	17-4-2	张森
● 定义模块接口	17-4-3	17-4-8	严灏
● 数据库	17-3-30	17-4-8	张森
● 各模块的具体实现算法	17-4-9	17-4-19	张森、严灏
● 制定模块测试方案	17-4-9	17-4-19	严灏
☐ ● 实现	17-4-20	17-5-3	张森、严灏
● 编写程序源代码	17-4-20	17-4-27	张森、严灏
● 模块测试和调试	17-4-27	17-4-30	严灏
● 编写用户手册	17-5-1	17-5-3	严灏
● 评审	17-5-1	17-5-3	张森
☐ ● 测试	17-5-4	17-5-17	张森、严灏
● 单元测试	17-5-4	17-5-8	张森
● 集成测试	17-5-9	17-5-12	严灏
● 验收测试	17-5-13	17-5-17	张森、严灏
☐ ● 维护	17-5-18	17-5-24	张森、严灏
● 完善错误应用	17-5-18	17-5-20	张森、严灏
● 编写故障报告以及修改报告	17-5-21	17-5-22	严灏
● 修订用户手册	17-5-23	17-5-24	严灏

甘特图：

4 支持条件

4.1 计算机系统支持

a. 硬件支持：

计算机：x86 架构个人计算机（对计算机本身没有太大要求）

b. 软件支持：

开发工具：QT Creator，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：

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

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“4.4.4 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）