

汇报小组：第九组

垃圾识别分类系统

项目总结报告

杨文豪，陈洪涛，侯同庆

2022-6-1

目录

1. 引言	2
1.1 编写目的	2
1.2 项目背景	2
2. 系统开发情况	2
2.1 开发环境	2
2.2 运行环境	3
2.3 开发结果	3
3 开发工作评价	4
3.1 对生产效率的评价	4
3.2 对产品功能的评价	4
3.3 对技术方法的总结	5
4. 项目流程	5
4.1 项目启动	5
4.2 需求阶段	6
4.3 设计阶段	6
4.4 开发阶段	6
4.5 测试阶段	7
4.6 验收阶段	7
4.7 项目开发流程各阶段产物	7
4.8 项目流程图	9
4.9 与计划工作的对比	10
5. 项目展示	11
5.1 主界面	11
5.2 功能界面	12
6. 风险管理	12
6.1 风险管理过程	12
6.2 风险管理相关表	13
6.3 风险登记册	15
7. 总结及经验教训	15
7.1 开发团队	15
7.2 需求的调研	15
7.3 做好开发计划	16
7.4 很好的沟通	16
7.5 做好工作总结	17
8. 展望（未来计划）	17
9. 课程总结	17

1. 引言

1.1 编写目的

垃圾识别分类系统的开发已经基本完成。本报告的目的是对垃圾识别分类系统进行项目的开发总结说明，以使用户及项目开发人员了解产品详细的设计与实现流程。为开发人员提供开发参考书。以及让我们在今后的项目开发中有更多的有据的资料来规范我们的开发过程和提高我们的开发效率。本报告的预期读者有用户、项目经理、开发人员以及跟该项目相关的其他人员。

1.2 项目背景

(1) 项目名称：基于 OpenCV 和 TensorFlow 的生活垃圾图像分类识别

(2) 任务提出者：杨文豪、侯同庆、陈洪涛

(3) 开发者：杨文豪、侯同庆、陈洪涛

(4) 选择背景：日常生活中的垃圾一般分为有害垃圾、可回收垃圾、干垃圾、以及湿垃圾这四类，对不同类别的垃圾应采取不同分类方法，如果投放不当，可能会导致各种环境污染问题。合理地进行垃圾分类即能提高垃圾资源处理效率，也能缓解环境污染问题，因次我们在图像识别的基础上设计一个垃圾分类系统，可以高效地实现对日常生活中常见垃圾进行智能识别分类

2. 系统开发情况

2.1 开发环境

操作系统：Windows 10

开发平台：PyCharm、Python3.8、Git

开发框架：TensorFlow 2.3.0

2.2 运行环境

操作系统：Windows 10

运行框架：Tensorflow 2.2

2.3 开发结果

1. 产品

- (1) 名称：基于 OpenCV 和 TensorFlow 的生活垃圾图像分类识别
- (2) 开发系统：垃圾识别分类系统

2. 项目开发文档

- (1) 项目计划书
- (2) 详细设计报告
- (3) 风险管理报告
- (4) 风险登记册
- (5) 项目验收报告
- (6) 项目总结报告

2.4 系统主要功能

- (1) 获取图像：从本地打开一张图片
- (2) 图像显示：在屏幕上显示当前已经获取的图像
- (3) 识别图像：使用预先训练好的模型对图像进行识别，得到图像中垃圾的名字
- (4) 垃圾分类：根据识别出来的垃圾名字，使用系统内部已经标签好的数据，对垃圾名字进行分类。
- (5) 信息展示：展示垃圾名字，并且显示当前垃圾的种类

3 开发工作评价

3.1 对生产效率的评价

- (1) 系统开发已历时大概两个月的时间
- (2) 开发的反复性比较多
- (3) 小组机器学习的开发基础不是很扎实

综合以上，此项目的开发效率不是很高，相反有相当一定时间的浪费。

3.2 对产品功能的评价

经过我们小组成员的共同努力协作，垃圾分类识别系统已经很好的完成了最初的业务需求。经过使用过程的观察，此项目开发的还是比较成功，但是还是存在着一些问题，造成这些问题的原因是多方面的。如：垃圾数据集的处理和部分代码的构建缺陷、模型的选取也存在一定问题，导致垃圾的成功识别率不是很高。但总的来说，此系统的功能开发还是一个

比较成功的案例。

3.3 对技术方法的总结

在此项目中使用到技术和工具：

1. Anaconda: Anaconda 里面包含众多常用的 python 开发包及其依赖, 包含 conda、python 等众多工具。使用 conda 可以对 python 包和环境进行管理。

2. Pycharm: Pycharm 待用一整套 python 开发过程中需要用到的工具, 如调试、语法高亮、项目管理、代码跳转、单元测试、版本控制等, 提高开发效率。使用 Pycharm IDE 可以更好的帮助程序进行编码、分析、管理等。

3. Tensorflow: Tensorflow 是一个端到端开源机器学习平台。它拥有一个全面而灵活的生态系统, 其中包含各种工具、库和社区资源, 可助力研究人员推动先进机器学习技术的发展, 并使开发者能够轻松地构建和部署由机器学习提供支持的应用。

4. Git: Git 是一个开源的分布式版本控制系统, 可以有效、高速地处理从很小到非常大的项目版本管理。

4. 项目流程

4.1 项目启动

- 产品经理和组内人员确定项目方向, 确定项目的主要执行人和管理人员等;
- 小组内部确认项目组团队组成, 包括产品经理、研发项目经理、研发工程师、测试人员等;
- 明确项目管理制度, 每个阶段的成果产物需要进行相应的评审, 从项目启动起, 研发项目经理每周提供项目研发汇报; 测试阶段, 测试人员每周提供项目测试汇报;

- 产品经理进行需求调研，输出《需求分析》文档，需求调研的方式主要有背景资料调查和访谈。
- 产品经理及组内成员对项目的业务进行梳理。首先，明确每个项目的目标；其次，梳理项目涉及的角色；再来，每个角色要进行的事项；最后，再梳理整个系统分哪些端口，要有哪些业务模块，每个模块再包含哪些功能。

4.2 需求阶段

- 进入可视化产物的输出阶段，产品经理提供最简单也最接近成品的系统原型，线框图形式即可。在这个过程中还可能产生的包括业务流程图和页面跳转流程图。业务流程图侧重在不同节点不同角色所进行的操作，页面跳转流程图主要指不同界面间的跳转关系；
- 产品经理面向整个小组，进行需求的讲解；
- 研发项目经理根据需求及项目要求，明确《项目里程碑》。根据项目里程碑，完成《项目计划书》，明确详细阶段的时间点，最后根据开发计划，进行项目任务分解，完成项目的分工；
- 项目成员按照各自的分工，进入概要需求阶段。

4.3 设计阶段

- 项目经理根据产品的原型，设计出系统界面效果图，并提供界面的标注，最后根据主要的界面，提供一套 UI 设计规范。UI 设计规范主要是明确常用界面形式尺寸等，方便研发快速开发。UI 设计常涵盖交互的内容；
- 研发工程师在界面效果图，输出《系统详细设计》，需求规格应包含最终要实现的内容的一切要素；
- 研发工程师根据《系统详细设计》完成正式编码前的一系列研发设计工作。

4.4 开发阶段

- 研发工程师正式进入编码阶段，这个过程虽然大部分时间用来写代码，但是可能还需要

进行技术预研、进行需求确认；

- 编码过程一般还需进行系统进行整体的联调等；
- 完成编码后需要进行功能评审；

4.5 测试阶段

- 测试人员按阶段设计《测试实例》，未通过的流程测试提交至分配给相应的开发人员调整；
- 研发人员根据测试结果修改代码，完成后提交测试，测试通过后完成；
- 测试人员讲解包括功能测试结果、压力测试结果等报告；
- 测试人员讲解系统各端口的操作方式、维护方式等。

4.6 验收阶段

- 在课程结束阶段，准备项目的汇报工作，并把系统进行试运行，稳定后开始最终项目汇报。

4.7 项目开发流程各阶段产物

表 1 项目开发流程产物表

阶段	负责人	时间	产物	成果描述
项目启动	杨，陈，侯	2022.4.1	《需求分析报告》	主要工作是确定项目方向，小组成员及组员之间的职责和分工
需求阶段	杨	2022.4.14	《项目计划书》 《风险管理报告》 《风险登记册》	对项目进行开发流程进行梳理及设计，并同时确定项目开发整个阶段的风险管理

设计阶段	陈，侯	2022. 4. 26	系统界面设计 (png)、 《详细设计文档》	设计系统的原型，并对项目进行详细系统设计
开发阶段	侯	2022. 5. 24	系统使用模型、 系统 demo	研发工程师正式进入编码阶段，这个过程虽然大部分时间用来写代码
测试阶段	陈	2022. 5. 25	测试样例	系统的测试样例全部通过后可进行下一阶段
验收阶段	杨	2022. 6. 1	汇报报告（PPT）、 《项目总结报告》、 《项目验收报告》	对项目进行验收工作，并准备项目的汇报

4.8 项目流程图

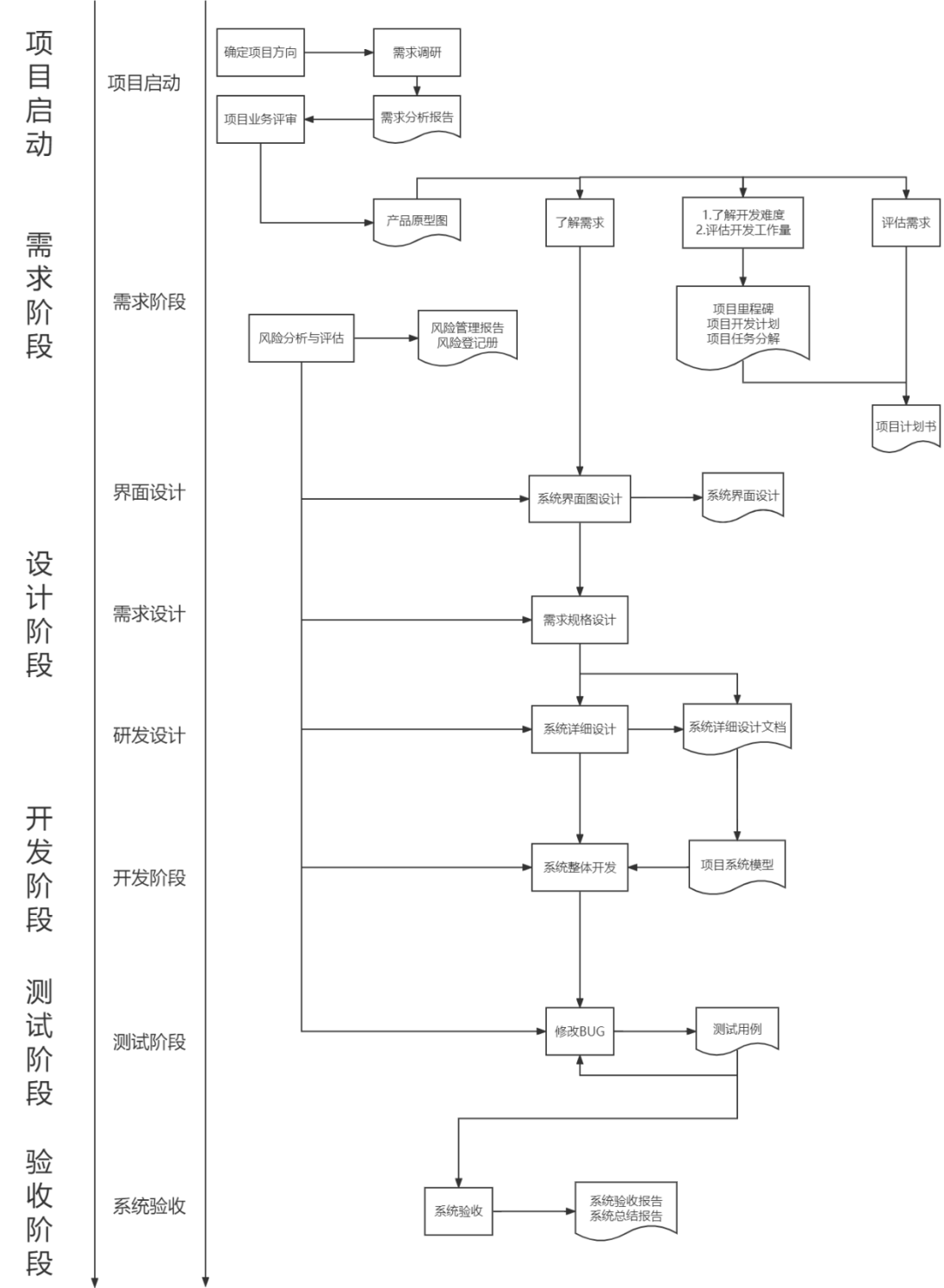


图 1 项目流程图

4.9 与计划工作的对比

(1) 图 2 为项目的里程碑趋势图，展示了每个阶段里程碑的实际完成时间和预计完成时间的对比

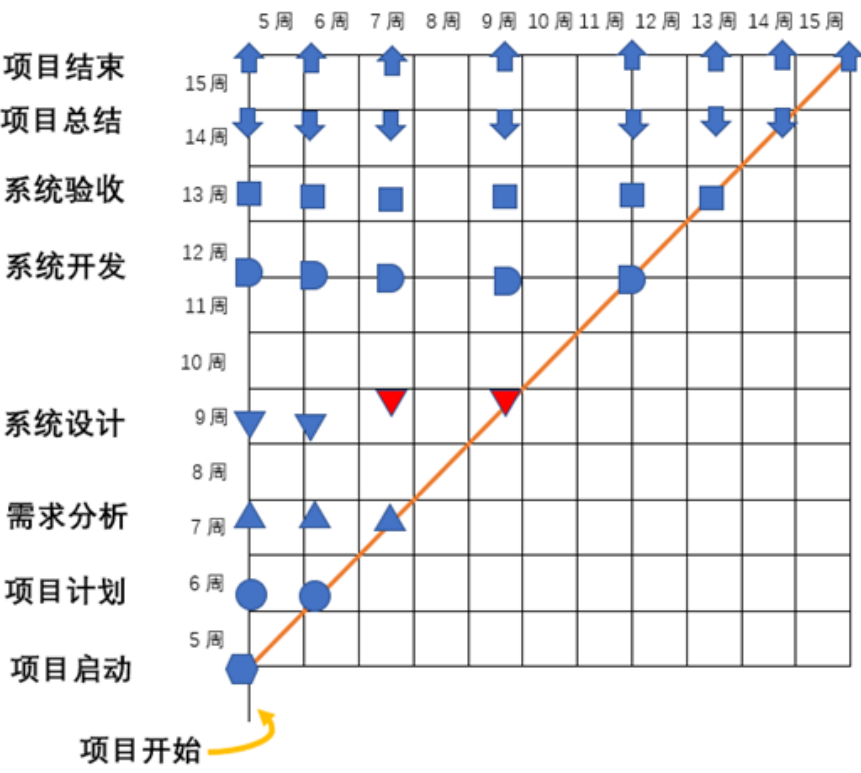


图 2 里程碑趋势图

(2) 图 3 为项目子任务完成时间和预期时间的对比图

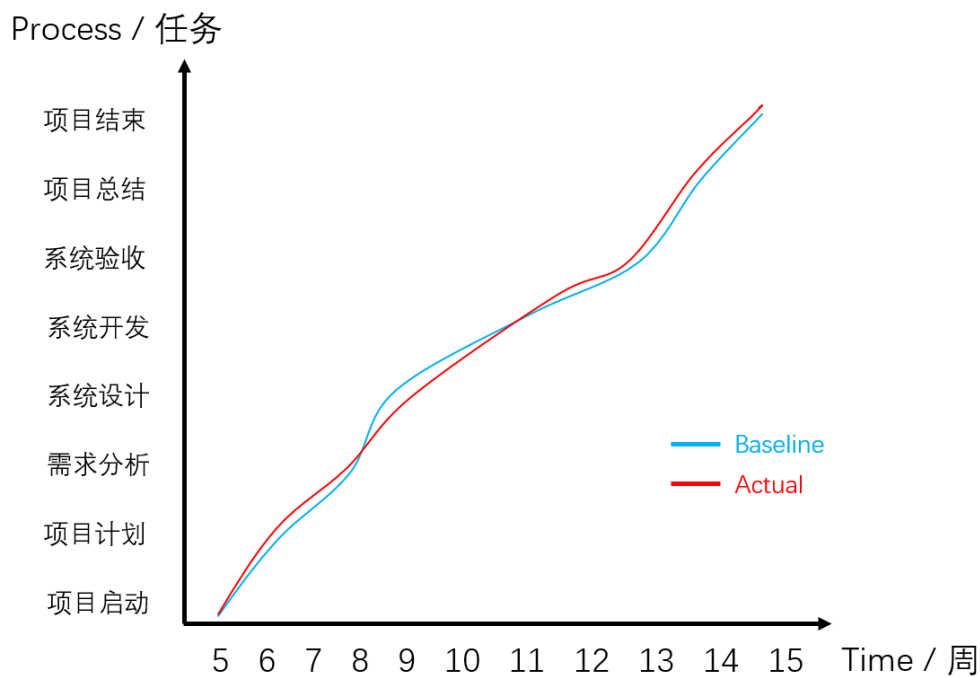


图 3 子任务完成时间对比图

5. 项目展示

5.1 主界面

打开软件，进入到图 4 主界面，可以看到相关信息。下面蓝色字体是 github 链接。



图 4 系统主界面

5.2 功能界面

在图 5 识别界面，点击上传图片，选择图片，点击开始识别得到图片的信息。包括垃圾的种类、物品名称和信息。



图 5 功能界面

6. 风险管理

6.1 风险管理过程

- 软件风险是指软件开发过程中及软件产品本身可能造成的伤害或损失。风险的最大特征是不确定性，也就是说它可能发生，也可能不发生。
- 风险管理在项目开发和管理过程中有非常重要的地位：
- 有效的风险管理可以提高项目的成功率。在项目早期就应该进行必要的风险分析，并通过规避风险降低失败概率，避免返工造成成本上升；
- 提前对风险制定对策，就可以在风险发生时迅速作出反应，避免忙中出错造成更大损失；
- 风险管理可以增加团队的健壮性。与团队成员一起做风险分析可以让大家对困难有充分估计，对各种意外有心理准备，不至受挫后士气低落；而项目经理如果心中有数就可以

在发生意外时从容应对，大大提高组员的信心从而稳定队伍；

- 有效的风险管理可以帮助项目经理抓住工作重点，将主要精力集中于重大风险，将工作方式从被动救火转变为主动防范。

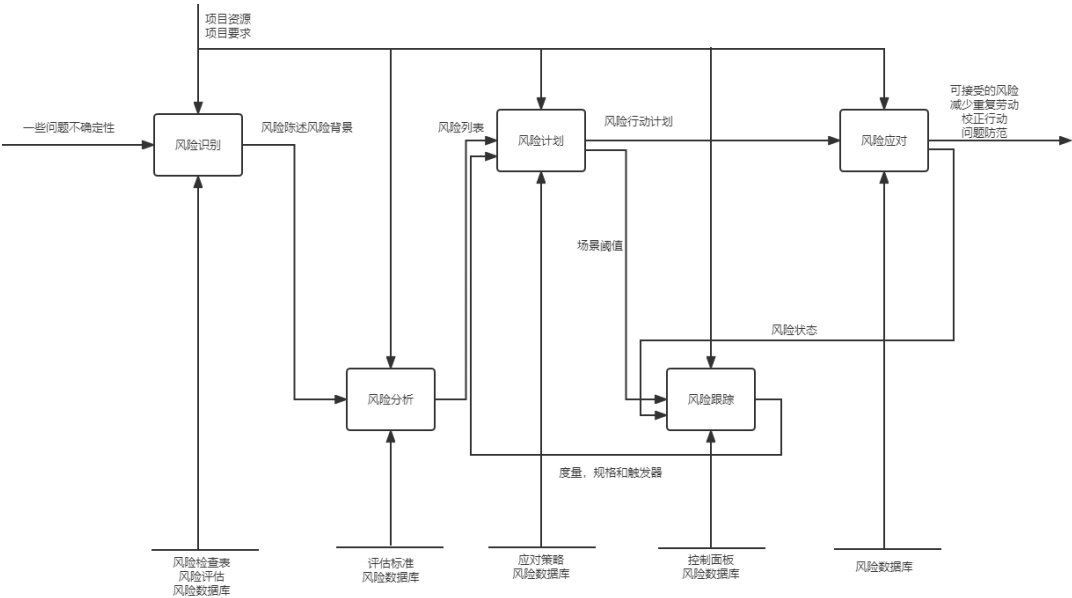


图 7 风险管理过程

6.2 风险管理相关表

(1) 风险管理过程中的有关的角色及职责

表 2 职责表

角色	职责描述
项目经理	负责风险管理的所有工作的执行，借助风险数据库和项目控制面板处理风险
开发人员	配合项目经理来识别和处理风险
设计人员	协助项目经理负责风险管理的所有工作的执行
管理人员	配合项目经理管理项目开发的各种相关工作

(2) 概率评估准则

表 3 概率评估表

等级（0-1.0）	编号	发生概率	等级说明
-----------	----	------	------

0.1	P1	0%-10%	既不可能发生
0.3	P2	11%-30%	发生的可能性很小
0.5	P3	31%-70%	有可能发生
0.7	P4	71%-90%	发生的可能性很多
0.9	P5	91%-100%	极有可能发生

(3) 风险后果评估标准

表 4 风险后果评估标准表

等级名称	代号	系统风险定义
可忽略	S1	对项目开发没有影响或影响很小，不影响项目开发周期
轻度	S2	对项目的开发有较小的影响，可通过调整进行解决
中度	S3	对项目开发的影响适中
重度	S4	对项目开发有重要影响，可能会导致项目的失败
致命	S5	会直接导致项目开发终止

(4) 风险评估表

表 5 风险评估表

后果	S5	15	19	22	24	25
	S4	9	14	18	21	23
	S3	6	10	13	17	20
	S2	3	8	9	12	16
	S1	1	2	4	7	11
	P1	P2	P3	P4	P5	

	风险发生概率
--	--------

6.3 风险登记册

见附件《风险登记册》

7. 总结及经验教训

7.1 开发团队

在项目确立后，要尽快的建立起项目开发团队。

项目团队成员的团结合作、相互沟通是非常重要的，团队成员之间要相互学习彼此的优点和技术，使团队的能力不断的提高。这样，在项目的开发过程中，团队才不会被难题困住不动。另外，团队中要有一个项目负责人，这个人无论是在与客户的沟通上，还是在技术上都要是很出众的人，此项目负责人要能很好的沟通客户与开发成员之间，以此来更好的理解客户的功能需求。人的记忆力总是有限的，所以就要求开发团队成员要尽量的书写一些开发文档，这些文档往往是我们在项目开发后期要用到的可寻资料。项目团队士气是项目成功的一个因素，我们需要不断的来培养我们的团队气势，使我们的团队不断的壮大。

7.2 需求的调研

在项目确立后，就到了需求调研分析阶段。

(1) 项目组对客户整体组织结构、公司有关人员的关系、职责等如果没有一个很好、足够的了解掌握，这样项目组就无法很好的完整的整理到客户的需求、或者说客户真实的功能需求，如此以来我们就为自己埋下了地雷，影响项目的开发周期，这就要求我们要与客户搞好无论是工作上的还是生活上的朋友关系，要深入的去了解客户需求。

(2) 我们要尽量的让客户也参与到项目的开发团队中来, 也就是说我们要使客户把自己也纳入到项目的开发团队中来, 如此一来, 我们掌握客户需求的真实性、可靠性就会大大的提高, 也就不会为项目的后期功能开发埋下陷阱。

(3) 在需求调研过程中, 如果缺乏足够用户参与, 这样的需求调研也是失败的。很多程序员不愿参与到客户的需求调研中去, 他们觉得与客户沟通不如与代码沟通容易有意思。尽管这样, 我们还是必须用足够多的时间去和客户进行沟通, 了解他们真实的需求。很多用户也是如此, 他们自己也不愿意参与到项目的需求调研中来, 虽然现状如此, 我们还是要努力的使客户参与到需求的调研中来。

(4) 在一个项目的开发中, 文档的书写是极为中要的一项工作。因为, 某些文档就是我们在开发后期与客户沟通的可寻依据、也是我们程序员在编码过程中要用到的重要文档。这就要求我们在需求调研中做好需求文档的记录和整理。

(5) 需求调研工具选择, 客户一般对图形还是比较感兴趣的, 所以我们在调研过程中, 我要尽量的采用图形化界面来和客户沟通需求。比如可以采用工具, 把客户的意思转换为用例图、时序图、协作图、状态图、类图等, 使表达的意思更加直观。这样客户会更快的进行问题的实质。

7.3 做好开发计划

在项目确立后, 我们就需要做好项目开发计划, 需求调研用时, 开发用时, 测试用时, 实施用时, 维护用时。在我们做好了计划后, 我们要随时跟踪计划任务的完成进度, 从而使我们的项目进度掌控在我们的开发周期范围之内, 今日计划、行动, 明日成功。

7.4 很好的沟通

在其他行业中, 人与人的之间的沟通只很重要的。项目开发也不例外, 很好的沟通能够加快项目的进度, 这就要求我们每一个开发人员要学会和善于沟通于客户和同事之间。在一个项目的开发过程中, 我们与客户的沟通是一个不断交流和沟通的过程。在开发到一定的阶段, 我们就需要和客户沟通已有功能, 尽量的去避免一些隐藏的问题, 及时的发现问题, 解决问题, 从而按时或者提前完成项目的开发。

7.5 做好工作总结

在项目进行的过程中，我们要不断去整理自己的工作情况和做好总结，这样以来，无论是在自己的技术还是其它方面，都会对我们有很大的提高，在长期的积累后，无论是我们个人能力，还是我们的团队能力都会有很大的提高。

8. 展望（未来计划）

该系统对于某些方面仍然有改进的地方，例如：

（1）由于目前系统只支持 PC 端使用，在使用时是极其不方便的，所以难免会导致系统的推广使用，并且图片的方式也大都只能通过智能手机拍摄，然后传入到 PC 端才能识别，所以将系统搬至移动端是必须要做到的；

（2）系统目前智能是处理静态图片，对于那些需要实时性要求很高的用户而言可能是非常不友好的，所以在第一阶段完成的前提下，我们小组期望能够在移动端的系统上加上实时性检测，具体操作大概为：用户在移动端上进行使用，在将摄像头对准检测目标之后，移动端界面会出现检测目标的相关信息名称，然后点击该名称，用户就可以得到该检测目标的垃圾分类信息。

9. 课程总结

本学期的《软件项目管理》课程除了培养了我们的项目管理的能力，还培养了团队合作相关能力。开发的这个系统，从设计到开发都经过了细致摸索和推敲和实地考察，这是我们小组共同努力的结果。系统开发中，大家的能力，诸如大家的合作，个人的协作能力，开发能力，以及时间观念都有一定的提高。希望软件系统的设计能给大家耳目一新的感觉，希望成为广受用户的欢迎。

通过本次的软件管理课程，我们也真正认清楚，一个成功的设计不应是自己“想当然”地去开发软件，而是应该以用户为出发点，始终在考虑“用户需要什么”，系统能够解决哪方面地问题，虽然小组成员都不是典型的用户，但开发者们也应该从用户的观点触发，去考虑如何设计系统地功能等。一味从自我考虑，只做符合自己的软件，最终就会发现它的需求是如此的不足，功能有很大的缺失，最后会发现做出来的软件连你自己的愿望。

同时，在项目开发过程中，我也深刻感受到了团队的重要性，以及团队管理的重要性，在现代软件开发过程中，一个人的“单打独斗”固然不是轻松的，依靠团队的力量，各司其职，团队的成员之间以高效的管理进行软件开发将是一次极其有效率的工作。当然，团队管理的重要性也不言而喻，没有好的团队管理，项目的开发工作将是一盘散沙，项目开发进度也会进行不下去，最终只能导致项目的失败。

这次的项目管理课程让我（们）认识到了很多的在实际开发工作中可能会遇到的事情，虽然不是一次正式的软件开发，但相信，这次的经历会给我（们）带来深刻的影响，并会称为以后真正的软件开发的“前车之鉴”。