**课程编号：C0801003232**

**软件质量保证与测试**

**作业报告**



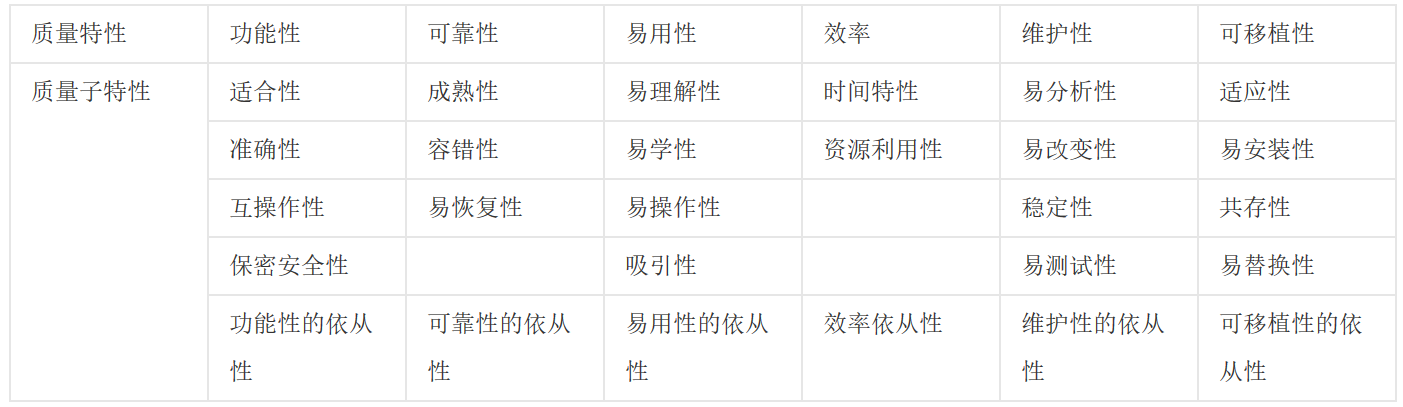
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **姓 名** |  | **学 号** | |  |
| **班 级** |  | **指 导 教 师** | | **吴辰铌** |
| **开 设 学 期** | **2018-2019秋季学期** | | | |
| **开 设 时 间** | **第2周——第17周** | | | |
| **报告日期** | **2018.12.22** | | | |
| **评 定 成 绩** |  | | **评定人** | **吴辰铌** |
| **评 定 日 期** |  |

**东北大学软件学院**

**1、主要质量标准和依据**

1. 质量标准

经过项目调研和资料查阅，“老来乐”手机应用的开发应满足软件质量通用标准，即经ISO/IEC 9126-1991标准改良版ISO/IEC 25010:2011**[1]**，该标准包括3层：质量特性、质量子特性和度量。其中质量特性包括功能性、可靠性、易用性、效率、维护性和可移植性。不同的质量特性拥有不同特征的质量子特性，如表所示：



下面将从不同的质量特性为产品定义质量标准：

1. 功能性——实现的功能达到设计规范并满足用户需求的程度。

在公司定位中，“无忧”公司是一个致力于服务老年人的高科技公司，研发水平需要达到开发出正确功能并满足规格说明的产品。由项目需求可得到，产品需要达到的功能要求包括至少如下5点：

* 1. 个性化健康资讯和健康咨询服务功能
  2. 手机定位监控运动轨迹功能
  3. 指定手机终端信息发送功能
  4. 语音操作自动进行医院预约挂号功能
  5. 智能过滤垃圾信息（欺诈短信、来电、视频、链接等）
  6. 帮助老年人理财功能

以上5点基本功能均需要满足适合性，在手机轨迹监控中需要满足定位准确性要求，智能过滤垃圾信息需要满足老人信息的保密安全性以防止对外泄露，理财功能需要满足良好的互操作性，考虑到用户为老人，理财界面和功能应该尽可能简单易用。

1. 可靠性——在规定的时间和条件下，维持其性能水准的程度。

既然用户定位是老人，在某些功能上的可靠性则需要有较高的要求，在老人正常的作息时间段内需要保持App的良好运行性能。

1. 易用性——用户掌握软件操作所要付出的时间及努力程度。

因为用户为60岁以上的老人，App对软件的易用性要求很高，以方便老年人的使用，具体量化标准为，用户掌握软件基本操作的熟悉时间不超过一小时。

1. 效率——软件执行某项功能所需的手机资源和时间的有效程度。

App响应时间不得超过5秒且必须有缓冲提示，必要时需要提供语音等待提示。App在手机中的耗电占比应尽可能低，手机定位监控运动轨迹的响应频率在2秒内，实时做到垃圾信息的过滤响应。

1. 维护性——当环境改变或者软件发生错误时，执行修改或者修复所作的努力地程度。

当App发生系统崩溃时，首先通过与语音或文字向老年人给出明显的提示，之后回滚当前事务并重启软件，重启软件的响应时间不应超过5秒。

1. 可移植性——从一个系统/环境移到另一个系统/环境的容易程度。

根据用户定位和市场调研，老人机系统通常为Kai OS，即App能在该系统上良好运行，为了满足其他用户的需要，该App同样要求能在Android和IOS系统上良好运行，以满足当前市场的主流智能机系统，为使用智能机的老人也能提供相同的服务。

1. 依据

ISO/IEC 9126 (1991) :软件产品评估—质量特性及其使用指南纲要。在此标准中，定义了六种质量特性，并且描述了软件产品评估过程的模型。

1、功能性的质量指标

功能的正确性：系统功能和用户的实际需求、已定义的产品规范一致。

功能的准确性：系统产生的结果在精度允许的误差范围内。

功能的完整性：所有功能及其定义清楚、可用。

2、可用性的质量指标

可操作性：容易使用和操作，包括理解用户界面、适应一些特殊用户的可选项等。

通用性：数据显示、网络通信接口和用户界面等都遵守已有的软件标准。

一致性：在软件开发整个生命周期内建立和使用相同的标准，保证全局变量、数据类型、出错处理的命名和使用一致。

3、可靠性的质量指标

自我恢复能力：当系统的某个功能失效发生时，系统在当前环境下能实现故障自动转移，重新自动配置、继续执行的能力，软件系统具有自我检测、容错、备份等机制，尽量做到独立于硬件的编码、硬件设备之间的通信协议一致等。

健壮性：各种恶劣环境（大数据量、大用户量）下系统能正常工作。

分布性：软件系统的某些子功能或子系统被定位于不同的处理主机、存储设备。

4、性能的质量指标

有效性：系统在通信、处理、存储等方面占有很少资源或者对所使用的资源进行了优化。

完整性：系统具有良好的安全管理，能防止不安全存取系统、防止数据丢失病毒入侵等。

易存取性：对系统的存取权限设置清楚，存取操作方便，存取操作有记录。

5、可维护性的质量指标

模块化：指讲一个复杂的软件系统分解为分别命名并具备最小耦合性、很强凝聚性、结构化的组件。

灵活性：容易为系统增加一个新功能或者新的数据而不需要进行大量的代码修改或者设计修改。

可测试性：测试软件组件或者集成产品时查找缺陷的简易程度。

可追溯性：对一个特殊需求容易找出相应的代码，反之，也可以根据代码找出特定的需求。

兼容性：软件、硬件、通信系统之间协调及兼容其他系统的能力。

　 可解释性：相关文档齐全、符合标准、逻辑清晰、描述准确、用词恰当，容易理解和定位。

6、可移植性质量指标

适应性：系统不依赖于环境，即系统不做修改或作很少的修改即可运行在其他环境下。

易安装性：与在指定的环境下安装软件所需努力有关的软件属性。如在线更新、安装包自动生成等。

可重用性：一个软件组件除了在最初开发的系统之外应用于其他系统的能力。

　　互操作性：软件系统与其他系统交换数据和服务的难易程度。

可替换性：与软件在该环境中用来替代指定的其他软件的机会和努力有关的软件属性。

质量就是客户满意的程度。实际上，软件的质量并非静态的而是动态的，假如客户对于某项质量指标没有要求，就没必要花费时间和工作去满足该项质量指标，因此，在具体软件系统的质量指标定义上，一定要结合用户的非功能需求，不但能恰到好处的满足客户需求，也能尽可能的节省开发成本。

**2、测试方法选择的影响因素及分析**

1. 影响因素

“老来乐”产品理念是：为全世界的老年人（年龄在60岁以上）提供安全、方便、高效、易用的健康生活服务应用。即产品非功能需求上需要达到安全、方便、高效易用的目标。为了更好的达到质量标准目标，需要选择合适的测试方法，根据项目理念分析，测试方法选择的影响因素包括：

1. 人为因素
2. 软件类型
3. （系统）环境因素
4. 错误类型
5. 具体分析
6. 人为因素

软件测试中的许多工作是由人来完成的，因此，人为因素是影响测试方法的一个重要方面。这样的差异并非偶然，这是因为[测试人员](https://link.jianshu.com?t=http:/www.ltesting.net/html/73/category-catid-273.html)在应用一个[测试方法](https://link.jianshu.com?t=http:/www.ltesting.net/html/news.html)时具有很大的灵活性，从而使测试人员的经验和学习能力可以有较大的发挥余地。所以只能采用越来越严格的测试方法和过程来有效地避免人为因素对测试方法和效率的影响。

1. 软件类型

软件类型也是影响测试效率的一个重要因素，而测试方法的选择又影响测试效率。即使是同一个测试者，利用不同测试方法测试不同类型的软件的效率和发现软件错误的能力也是不尽相同的。测试什么类型的软件的效率较高则因人而异，这往往与测试者的相关知识、经验与专业训练有关。

由于本软件的类型为App，针对该App的测试方法包括如下几种**[3]**：

* 1. 安全测试
     1. 软件权限
     2. 安装与卸载安全性
     3. 数据安全性
     4. 通讯安全性
     5. 人机接口安全测试
  2. 安装卸载测试
     1. 安装测试
     2. 卸载测试
  3. UI测试
     1. 导航测试
     2. 图形测试
     3. 内容测试
  4. 功能测试
     1. 运行
     2. 应用的前后台切换
     3. 免登陆
     4. 离线浏览
     5. App更新
     6. 定位、照相机服务
     7. PUSH测试
  5. 性能测试
  6. 交叉事件测试
  7. 兼容测试
  8. 回归测试
  9. 用户体验测试
  10. 手势操作测试
  11. 客户端数据库测试

1. （系统）环境因素

App所存在的系统环境也是影响测试方法的因素之一。考虑到App供老人使用，其使用的系统大部分是老人机系统Kai OS,因此测试方法的选择上应该尽可能考虑手机的性能。

1. 错误类型

不同测试方法发现不同类型错误的能力也有所不同。错误类型的划分方法有多种。其中之一是把错误分成遗漏错误和多余错误：遗漏错误是指软件的实现中遗漏了一些处理和操作;多余错误是指代码中包含了不必要的处理和操作，而导致软件错误。还有一种分类是：初始化错误、控制错误、数据错误、计算错误、界面错误。统计数据表明，[功能测试](https://link.jianshu.com?t=http:/www.ltesting.net/html/93/category-catid-93.html)发现软件中的初始化错误、控制错误和数据错误的能力比其他方法高，而[静态测试](https://link.jianshu.com?t=http:/www.ltesting.net/html/90/category-catid-90.html)方法发现计算错误、界面错误的能力则比其他方法强。

**3、测试风险分析**

（1）测试风险

由项目需求分析得到，“老来乐”App的适用人群较为特殊，普遍为60岁以上的老人，所以在测试中面临的风险除了需求的变化导致功能改变引起测试方法技术选择改变的风险之外，还需要考察一系列其他针对性的测试风险并设定解决方案以应对App开发过程或上市后的变化。在此列出以下10条在软件测试中可能发生的风险**[4]**：

1. 需求风险
2. 测试用例风险
3. 缺陷风险
4. 代码质量风险
5. 测试环境风险
6. 测试技术风险
7. 回归测试风险
8. 沟通协调风险
9. 研发流程风险
10. 其他不可预计的风险

（2）具体理由

1. 需求风险

目前产品的基本功能性需求包含以下5点：

（1）能够为老人推送个性化的健康资讯和健康咨询服务。

（2）能够监控老人的运动轨迹，并能将信息发送给指定手机终端，防止老年人意外走失。

（3）能够通过语音操作，帮助老人在医院进行预约挂号。

（4）能够对欺诈信息、来电、视频、链接等进行智能过滤。

（5）能够帮助老年人理财。

非功能性包括：安全、方便、高效、易用

如果产品在开发过程中需求发生变更或理解不到位，产品测试范围将存在较大误差，遗漏部分需求或者执行了错误的测试方式；另外需求变更导致测试用例变更，测试用例维护成本增加，实时更新时存在误差。如果需求分析阶段处理不当可能会增加测试成本和测试人员的负担。

1. 测试用例风险

针对于本项目，测试用例在设计时首先应考虑测试方法，如黑盒或者白盒测试，根据不同的测试方法来按正确的步骤分析，最后给出测试用例，如果在设计时测试员经验不足或者忽视了某些如边界条件、异常输入的情况，则可能会为产品交付后所出现的崩溃承担一定的责任。

此外，测试用例设计不完整，忽视了边界条件、异常输入等情况，用例覆盖率没有做到足够覆盖，测试用例没有得到全部执行，有些用例被有意或者无意的漏测，需求变更导致的测试时间被压缩等情况。

1. 缺陷风险

如测试人员在本项目中的缺陷跟踪不够积极主动，没做好缺陷记录和及时更新，同样的缺陷，导致的原因可能不同，则会带来不可避免的缺陷风险。

1. 代码质量风险

该App如果由经验不足的程序员开发，有可能出现代码质量差，可读性差，重构性差，没做好注释等原因导致缺陷较多，修改难度增大；另外还有系统架构设计的不足，导致的扩展性不足，性能兼容差等问题，将为测试找出这一问题增加较大的负担。考虑到该公司是一家高科技公司，出现该风险的可能性较小，但是仍然需要考虑此种风险。

1. 测试环境风险

“无忧”公司的测试环境和生产环境配置不同，测试环境交叉影响较大，测试环境数据量不足导致的测试结果误差等问题。

1. 测试技术风险

由公司理念和产品定位可知，该项目可能存在技术难度，测试能力和经验所限，技术水平相对较差等问题导致测试进展缓慢，测试结果准确性不够，项目发布日期延期等风险。考虑到该公司是一家高科技公司，出现该风险的可能性较小，但是仍然需要考虑此种风险。

1. 回归测试风险

回归测试，一般时间相对来说较少，且大多只回归主要的功能点用例，可能造成漏测；另外还有回归验证缺陷时业务流走不通导致的打回修复再验证造成的时间延后问题。

1. 沟通协调风险

项目进行过程中需要多方沟通协调，不同部门，岗位之间的沟通、协作，难免存在误解、沟通不畅的情况，比如需求变更没有及时沟通，开发代码提交没有及时告知，测试结果的反馈不及时等问题。

1. 研发流程风险

产品研发的流程包括从产品需求评审、研发设计、代码提交、测试发布等一系列流程，流程的不规范不协调很可能导致很多问题如开发在不告知其他成员的情况下提交代码，发布没有预生产环境，生产出现问题无法及时回滚等情况。

1. 其他不可预计的风险

一些突发状况、不可抗力等也构成风险因素，且难以预估和避免。比如公司核心开发人员跳槽，需求方取消部分功能等情况，对于此类情况，往往一时无法解决，建议做好备份方案和容灾机制，或者采用灰度发布等措施。

（3）引入新工具后的分析

如果在测试过程中公司更换了新的配置工具，该风险属于其他不可预计的风险，如果配置管理工作发生变更，如新的配置管理工具的更换，会对其他的测试风险造成一定的影响，还可能增加新的软件测试风险，其中影响最大的风险包括测试环境风险、缺陷风险等。

由于不同（版本）的操作系统、不同（版本）的数据库，不同（版本）的网络服务器、应用服务器，再加上不同的系统架构等组合，使得要构建的软件测试环境多种多样、不胜枚举；而且随着软件运行环境的多样性、配置各种相关参数的“浩大工程”和测试软件的兼容性等方面的需要，使得构建软件测试环境的工作变得较为复杂和频繁，而软件测试环境的构建是否合理、稳定和具有代表性，将直接影响到测试结果的真实性、可靠性和正确性。

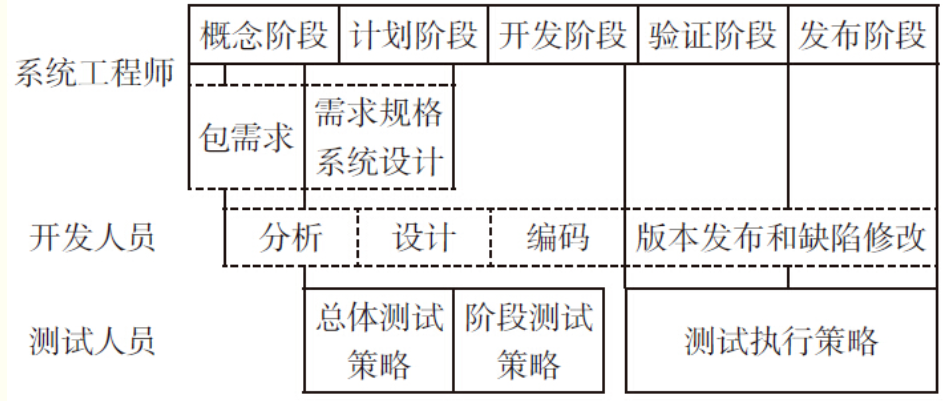
因此，一开始需要选择并确定好配置管理工具对App的测试至关重要，如果中途发生变更，不管是以前数据的备份和工具接口的对接都将对项目带来不必要的负担，增加工作量，最终的后果可能导致交付时间的延迟或隐藏的软件缺陷没被发现，使得产品维护变得异常困难。

最后在此给出测试过程中配置管理问题的建议解决方案**[5]**：

1. 选取合适的配置管理工具
2. 整理配置项，明确相应管理流程
3. 将配置项作为一个整体进行配置管理
4. 增加发布前验收测试环节
5. 采用并行开发方式区分不同的开发活动
6. 定制文件开发方式
7. 明确角色与职责

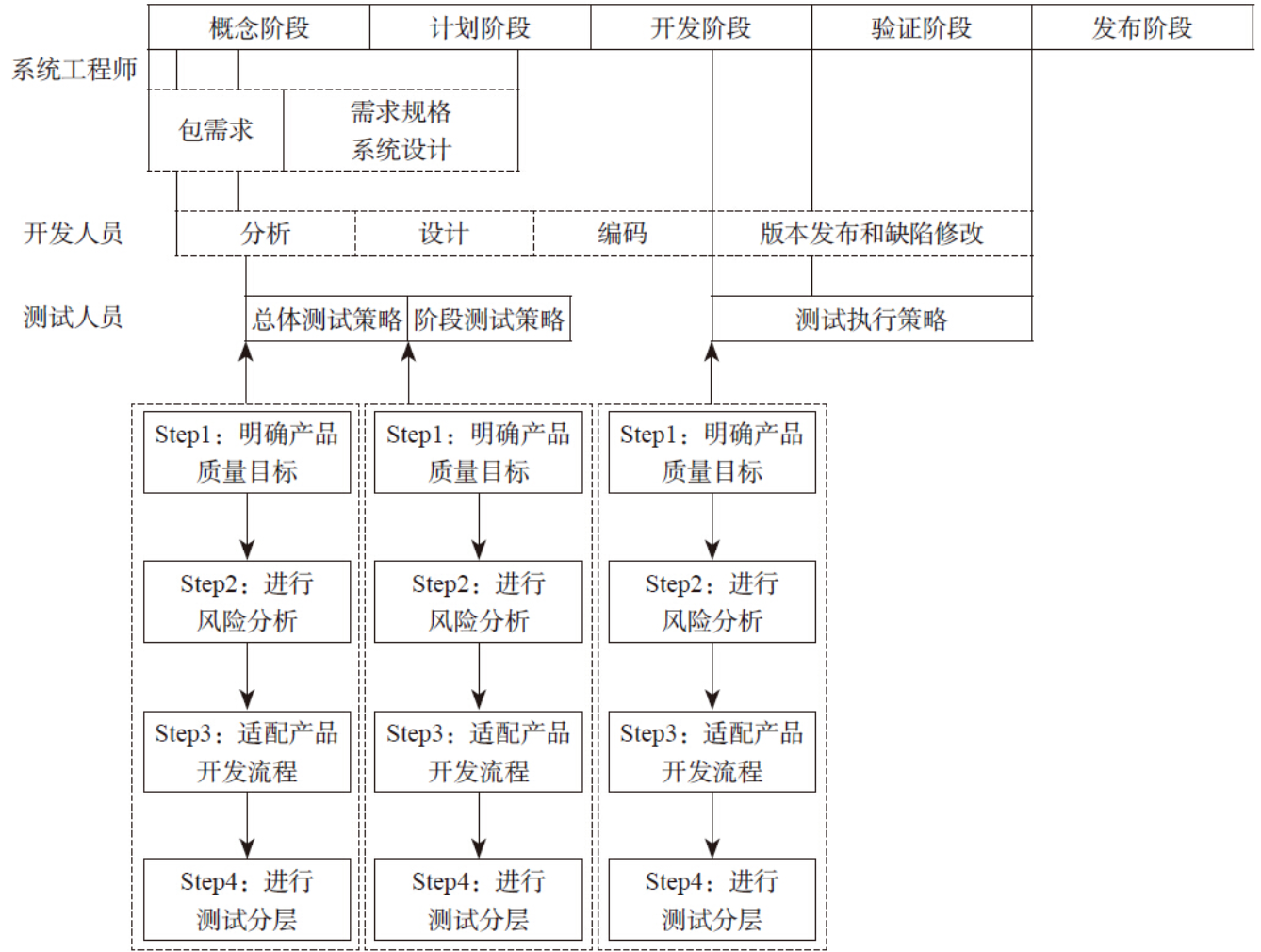
**4、总体测试策略**

在产品的实际开发流程中，首先需要明确总体测试策略结构所在的时间段，如下图所示：



所以目前项目需要制定的总体测试策略——包括确定产品质量目标，进行项目整体的风险识别，从产品层面来确定测试重点和测试难点、测试深度和测试广度，是测试策略的总纲。总体测试策略从概念阶段开始，在计划阶段前期完成比较合适。因为这时产品的需求、质量目标和整体形态都已经确定下来，已具备了制定总体测试策略的条件，而且也需要这样一份文档来总领后面的测试活动，让测试团队成员心中有数。

为了更规范的制定总体测试策略，项目需要按照测试策略的结构，在项目的不同阶段多次使用四步测试策略制定法来制定测试策略，如下图所示：



根据刘琛梅的《测试架构师修炼之道：从测试工程师到测试架构师》**[6]**的3.2章——像软件测试架构师一样的思考，将从以下7个方面分析，最终得出项目的总体测试策略。

（1）测试范围

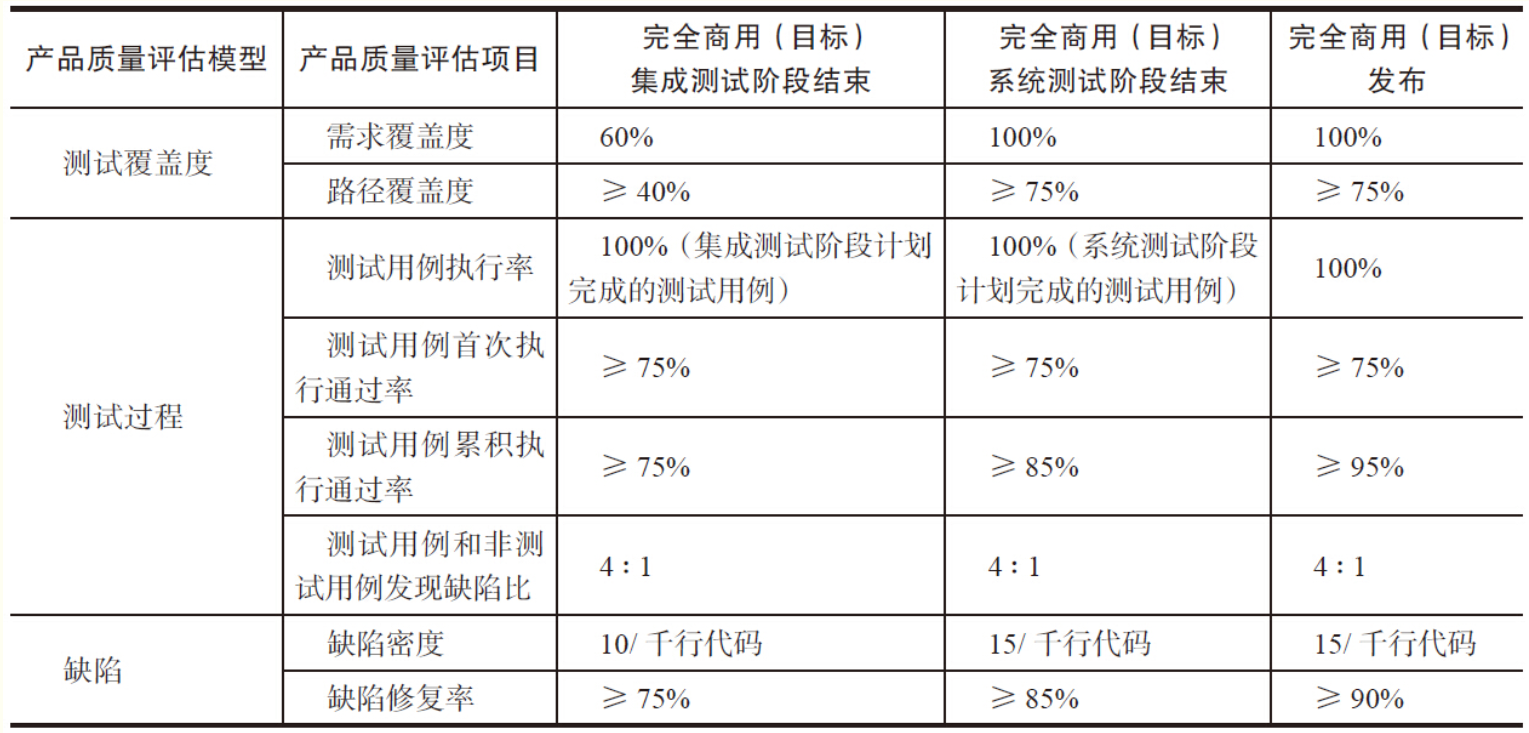
由产品的质量标准可知，项目的测试范围包括“老来乐”App的功能性和非功能性需求的测试，由产品的定位确定主要的非功能性需求的测试范围——安全、方便、高效、易用，分别对应质量特性中的功能性的质量子特性——保密安全性、可靠性、效率和易用性。

测试范围具体展开为：

1. 功能测试
   1. 个性化健康资讯和健康咨询服务功能
   2. 手机定位监控运动轨迹功能
   3. 指定手机终端信息发送功能
   4. 语音操作自动进行医院预约挂号功能
   5. 智能过滤垃圾信息（欺诈短信、来电、视频、链接等）功能
   6. 理财功能
   7. 服务器功能
2. 系统测试：需求中明确的业务流程，组合不同功能模块形成大的功能，如手机定位发送轨迹信息到指定手机终端功能，语音操作进行挂号功能。
3. 安全测试：
   1. 软件能否防御恶意信息注入攻击
   2. 软件能否自动识别垃圾信息并将投放账号放入黑名单
   3. 服务器能否防御恶意攻击
4. 性能测试
   1. App响应时间测试
   2. App耗电占比及内存使用测试
   3. 定位监控运动轨迹的响应频率测试
   4. 服务器峰值运转测试
   5. 并发性能测试
   6. 疲劳测试
5. 易用测试
   1. App基础功能掌握时间范围测试
   2. 操作步骤环节测试，操作流程是否简单

（2）测试目标

软件测试的目的是尽可能多的发现程序中存在的错误，根据产品定位，该App是一款需要投入市场的商用软件，在此给出商用软件的通用测试目标指标，如下表所示：



（3）测试重点

1. 产品价值

“老来乐”App的产品理念是为全世界的老年人（年龄在60岁以上）提供安全、方便、高效、易用的健康生活服务应用。其功能价值必须满足为老年人提供更加方便易用的基础服务，为老年人带来高质量的晚年生活。其非功能价值必须让老年人感到产品的易用性和易操作性，针对适用人群特点开发他们认为好用的产品。

1. 质量目标

由质量标准分析，该App的质量目标首先需要满足客户方便、高效、安全、易用的需求。即满足质量标准中的保密安全性、可靠性、效率和易用性。

1. 产品实现

该App开发公司是一家高科技公司，为了更贴合满足用户需求的产品，产品开发的新写代码应该大于60%，继承代码不超过30%，开源代码占10%左右。

1. 历史测试情况

根据不同阶段的测试目标，记录测试日志，从测试日志中发现并完成前阶段未完成的测试目标，在不同的测试类别上应按优先级测试，功能性测试的优先级最高，非功能性测试的优先级其次。

（4）测试难点  
 项目利用白盒测试技术需要达到以下功能性的代码测试要求，该部分测试过程较复杂，需要投入大量时间和精力才能完成所有功能代码的阶段性测试：

1. 健康咨询信息个性化推荐

该功能应该满足的技术条件是运用基于内容的推荐算法和基于用户的协同过滤算法（UserCF）找到用户感兴趣的信息。

1. 监控老人运动轨迹并将信息发送给指定手机终端

该功能推荐A-LBS SDK实现，该功能集成了手机定位的轨迹记录API，提供立即定位、批量定位、轨迹查询功能，对移动终端无要求、对GPS无要求、对手机操作系统无要求。A-LBS SDK由API和移动端SDK组成（目前支持android、ios操作系统，非智能机及其他操作系统建议使用LBS API）。

1. 语音操作帮助老人进行自助挂号

通过RNN卷积神经网络实现语音识别处理技术。

1. 智能过滤垃圾信息

通过关键词过滤或通讯信任技术实现此功能。

1. 理财服务

自主开发功能或理财公司合作开发功能。

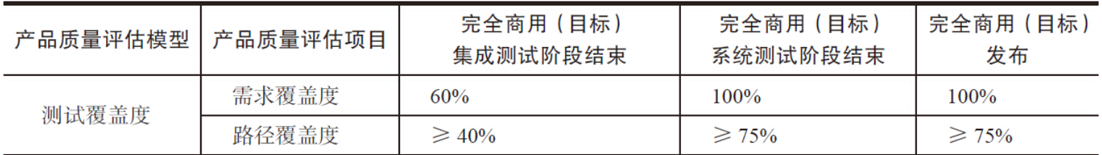
利用如下测试技术达到产品非功能性测试需求：系统测试、安全测试、性能测试、易用测试等，具体测试内容已在前面有介绍。

项目利用黑盒测试技术主要测试App的界面、基本功能和业务流程是否可以正常实现和运行。

（5）测试深度

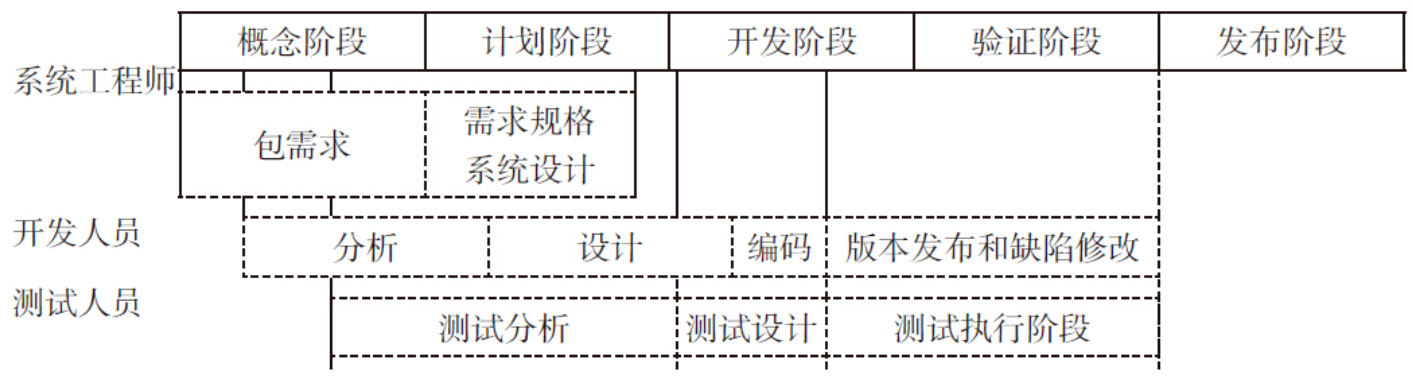
1. 静态测试：测试产品说明书
2. 开发实现：需要深入逻辑层
3. 单元测试：不需要深入函数级
4. 集成测试：需要进行应用系统的各个部件的联合测试
5. 系统测试：对整个产品系统进行功能性测试
6. 验收测试：部署软件前最后的测试
7. 性能测试：需要进行负载测试和压力测试
8. 边界值测试：需要加入健壮性测试
9. 安全测试：测试系统在防止非授权的内部或外部用户的访问或故意破坏等情况时做出的响应
10. UI测试：需要对界面的操作进行测试
11. 深度测试：执行一个产品的一个特性的所有细节

（6）测试广度

由测试目标可得出产品在不同阶段的测试广度要求，如下表所示：

（7）测试活动安排

测试策略中具体的内容，也需要和研发流程保持一致，项目只有根据研发流程来安排测试活动，才能确保测试和开发的节奏彼此吻合。测试在各个阶段的活动和开发的活动是能够配合起来的，如图所示：



为了做好下一阶段的版本测试策略，要求测试架构师能够和开发人员进行有效沟通，使得双方能够理解彼此的节奏，达到更好的配合。

**5、测试充分性判断**

项目从伦立军等人的《软件测试充分性研究》**[7]**了解到，软件测试充分性准则是决定一个软件系统能否被充分测试的规则，而测试充分性的关键问题是其检测故障的能力。在此项目从功能性测试和性能测试分析其充分性判断标准，为测试人员提供具体详细的测试充分性度量标准。

（1）功能性测试充分性判断标准

充分性准则的分类有各种不同的方法，基于功能性测试的充分性准则称为程序与需求项结合的充分性准则，本App测试要求需求测试的覆盖率为100%，该App所要要求达到的基本功能点和各自的测试充分性判断标准如下：

1. 个性化健康资讯和健康咨询服务功能

白盒测试：针对该部分代码的推荐算法如基于内容的推荐和基于用户的协同过滤的算法要求满足路径覆盖或条件组合覆盖或语句覆盖。要求通过根据独立路径所设计的测试用例。

1. 手机定位监控运动轨迹功能并发送定位至指定手机终端

白盒测试：针对该部分推荐A-LBS SDK调用的API代码要求满足路径覆盖或条件组合覆盖或语句覆盖。要求通过根据独立路径所设计的测试用例。

1. 语音操作自动进行医院预约挂号功能

白盒测试：针对该部分RNN算法的实现代码要求满足路径覆盖或条件组合覆盖或语句覆盖。要求通过根据独立路径所设计的测试用例。

1. 智能过滤垃圾信息（欺诈短信、来电、视频、链接等）功能

白盒测试：针对该部分关键词过滤或通讯信任技术实现的代码要求满足路径覆盖或条件组合覆盖或语句覆盖。要求通过根据独立路径所设计的测试用例。

1. 理财功能

白盒测试：针对该部分开发的代码要求满足路径覆盖或条件组合覆盖或语句覆盖。要求通过根据独立路径所设计的测试用例。

从黑盒测试的角度，其所有的功能都要求通过根据正确详细的等价类划分所设计的单缺陷或多缺陷测试用例，在此基础上加以健壮性测试的边界值分析，并符合预期输出结果。

从测试数据集的角度，为了使设计用例尽可能符合标准和规范，所有测试数据的选择应该满足测试数据选择准则形式的充分性准则**[8]**：一个测试数据充分性准则C是一个被测程序p及其功能规约s到一族测试数据集合的映射，即C：P\*S—>2T．t∈C(p，S)表示测试数据集合t对于测试程序p(相对于功能规约s)来说是充分的，否则是不充分的。

（2）性能测试充分性判断标准

1. App响应时间测试
   1. 测试数据满足如上功能性测试数据选择的充分性准则。
   2. 设计测试用例预期结果：App响应时间不得超过5秒且必须有缓冲提示，必要时需要提供语音等待提示。
2. App耗电占比及内存使用测试
   1. 测试数据满足如上功能性测试数据选择的充分性准则。
   2. 在高频长时间使用下测试耗电情况，测试用例不得少于20个。
   3. 设计测试用例的预期结果：App在手机中的耗电占比应不超过10%。
3. 定位监控运动轨迹的响应频率测试
   1. 测试数据满足如上功能性测试数据选择的充分性准则。
   2. 设计测试用例的预期结果：手机定位监控运动轨迹的响应频率在2秒内。
   3. 测试用例正确充分，测试数量不得低于所有测试用例的15%。
4. 服务器性能指标测试
   1. 测试数据满足如上功能性测试数据选择的充分性准则。
   2. 从以下方面考察Web服务器运行指标来进行测试，测试用例要求覆盖率达到90%：
      1. Avg Rps: 平均每秒钟响应次数=总请求时间 / 秒数；
      2. Avg time to last byte per terstion （mstes）：平均每秒业务脚本的迭代次数
      3. Successful Rounds：成功的请求；
      4. Failed Rounds ：失败的请求；
      5. Successful Hits ：成功的点击次数；
      6. Failed Hits ：失败的点击次数；
      7. Hits Per Second ：每秒点击次数；
      8. Successful Hits Per Second ：每秒成功的点击次数；
      9. Failed Hits Per Second ：每秒失败的点击次数；
      10. Attempted Connections ：尝试链接数。
5. 并发性能测试
   1. 测试数据满足如上功能性测试数据选择的充分性准则。
   2. 逐渐增加负载，直到系统的瓶颈或者不能接收的性能点，通过综合分析交易执行指标和资源监控指标来确定系统并发性能和最大服务级别。
   3. 测试用例封盖率应不得低于所有测试用例数量的20%。

**6、参考文献**

[1].https://blog.csdn.net/bmzydream\_007/article/details/5135410?tdsourcetag=s\_pctim\_aiomsg

[2].https://baike.baidu.com/item/%E8%B4%A8%E9%87%8F%E6%A0%87%E5%87%86/5996098?fr=Aladdin

[3]. <https://www.jianshu.com/p/966adb409f7f>

[4].https://baijiahao.baidu.com/s?id=1568255143029270&wfr=spider&for=pc

[5]. https://www.ibm.com/developerworks/cn/rational/r-cn-testprocessscm/

[6]. 《测试架构师修炼之道：从测试工程师到测试架构师》. 刘琛梅著. 北京神州绿盟科技有限公司

[7]. 伦立军，赵辰光，丁雪梅，李英梅.软件测试充分性研究[A]2004.40(3)

[8]. http://www.docin.com/p-901838522.html?docfrom=rrela

评价表格：

|  |  |
| --- | --- |
| **考核标准** | **得分** |
| （1）报告格式规范、语言通顺（10%）； |  |
| （2）质量标准分析合理（10%）； |  |
| （3）测试方法分析全面、合理（20%）； |  |
| （4）测试风险分析全面、合理（20%）； |  |
| （5）测试总体测试策略全面、合理（30%） |  |
| （6）测试充分性判断标准合理（10%） |  |