　　　　　　　　　　　　　　　　　　Android开发流程

1. 日程安排：预估项目完成需要的时间。划分具体项目的级别，大项目，中项目。项目的名称，项目预定时间，实际时间，延迟时间，完成进度，耗费工时，担当人，完成的状态。
2. 担当者一览表：担当人，开始时间，结束时间，式样。
3. 标准工程：具体项目分类，及工作完成时间。基本计划，设计，程序设计，试验
4. 设定：休出勤设定，书写设定，
5. 完成进度表.
6. 开发软件概要： a. 订货原委托内容：软件的功能要求对应，图文对应。

b. 项目等级判定

c. 成果物：如开发计画书，机能式样书，详细式样书，单体测试，结合测试式样书&结果报告书，综合评价计划书，报告书等等。

d. 要件管理：

e. 要件变更管理：就是开发的软件要求，发生变更时的管理情况。

f. 安全保障贸易审查方针：安全信息的保管

g. 注意事项：开源软件的使用

1. 日程：估计的时间，工时。

Android的代码规约

1. 包名：必须以com.brother开始，后面+项目名，模块名，层级
2. 类：采用驼峰规则，首字母大写，且为名词。
3. 方法名：为一个动词，首字母小写后面的大写。

单个返回值，使用get前缀，列表使用find前缀。

设置方法，使用set前缀。

布尔型判断，使用is,has前缀，或者是逻辑的单词如equals。

1. 变量名：首字母小写，后面大写。

布尔型变量：使用is前缀

常量命名：全部大写，使用下划线分割

异常变量：以Exception结尾

资源变量：资源名+控件类型全称

Layout文件：全部小写，下划线分割。图片按照可按下，不可按下划分

Id:资源名称+控件全称

1. 常用控件缩写：

editText ： editxt

imageView： imgv

button： btn

textview： txtv

listview ：listv

imagebutton： imgbtn

checkbox ：chkbox

radiobutton：radiobtn

1. 代码注释：代码注释需达到20%
2. Copyright说明：在类的顶部追加copyRight信息
3. 函数说明：函数一句说明，params说明参数，返回值，异常等等
4. 类成员变量和常量注释，使用java doc的形式
5. 方法内部的注释：不能使用java doc格式，可以使用//，/\*….\*/
6. 最好不要使用全局变量.方法或者构造参数不超过4个
7. NULL CHECK:在函数名和参数中使用@NonNull,@Nullable.
8. 使用Lombok,在需要外部访问@Getter,设置@Setter,资源自动释放@Cleanup,使代码不冗余。
9. If嵌套不得大于三层
10. 异常处理：如bitmap的oom

Android的开发票的实现

1. M票的是否在现：是，写再现手顺。否，写推测再现手顺，填写再现率。
2. 填写原因栏：1.技术观点，2.用户观点、3.分析人，原因分类等等。
3. 对应栏：对应策略，对应人和日期，分类，
4. 检索方法栏：检索方法
5. M票标题：追加xxx画面，等的填写
6. 问题栏：
7. 原因栏：
8. 对策栏：
9. 检查栏
10. M票修正时的影响范围考虑
11. M票修正的联络流程
12. 不再现的M票
13. M票修正速度的提高
14. 询问的方式：询问时说清楚看了那些资料，不清楚的地方在哪里
15. 做预算的注意点：①提高预算精度：明确了要求品质、实现方法后，再做预算

②被要求在短时间内作出预算的话，可以先做一个概算

③概算时，即使不明确实现方法，也OK。但是请写明“是在不明确实现方法的前提下做的”

④对要求品质，如没有特别的要求的话，则为该工程作业的通常要求

Android 测试观点

1. 参量
2. 条件：语言，国别，规格，，，，优先操作，时间（时差，闰年等），有效/无效 ，成功/失败，例外：功能式样之外的错误，组合：动作，条件，状态的组合。
3. 操作：

状态转换：根据上一次的设置，来进行下个场合。

中断：客观原因动作异常，中止，重新开始。

取消：发生异常后，继续，异常解除后，再次继续操作

顺序变更：操作顺序改变。

时间出界：设定一个最小时间

有效无效：未注册状态登录解除操作等

成功失败：机能成功的条件状态，和失败的条件状态

例外：现实几乎不发生的条件和状态

组合：动作，条件，状态的组合

1. 多功能：

同时：再做一个操作的时候，进行另一个操作。

过度：在一个操作中，进行下一个操作。

处理切换：在进行一个操作的时候，切换到另一个操作，再切换回来

中断：处理一个操作的时候，另一个操作挤进来。中断测试

合作：有一个功能让另一个功能产生影响的动作。。

组合：多个功能让另外的功能产生影响的操作

相互作用：确定应用程序的数据兼容

1. 资源：

同时：使用一个资源，在使用另一个资源操作

竞争：多台机器同时使用一个资源

1. 安全性：是否有危险性
2. 性能：是否符合规格
3. 障碍：系统故障
4. 负荷：
5. 构成：硬件，软件结构
6. 环境：外部环境
7. 保守：基本功能的确定
8. 可用性：效率，认知，一贯性，