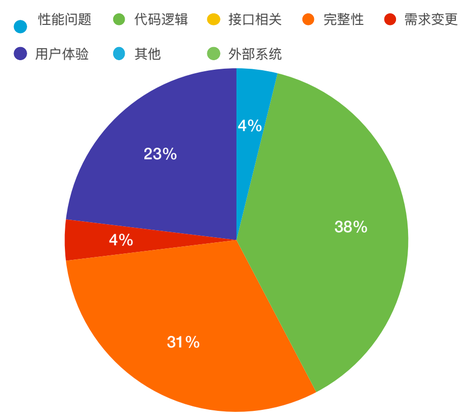
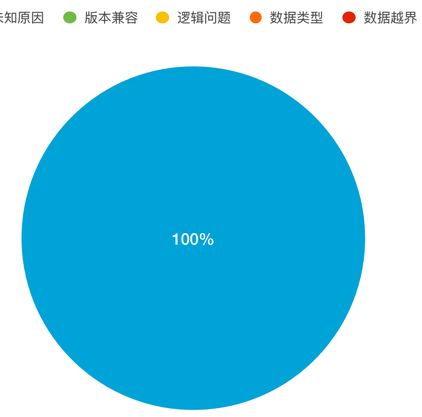
BUG类型

BUG类型	数量	BUG类型细分	数量
性能问题	1	数据越界	-
		数据类型	-
		逻辑问题	-
		版本兼容	-
		未知原因	1
代码逻辑	10	逻辑错误	6
		网络机制	4
接口相关	0	接口问题	-
		环境问题	-
		请求参数问题	-
完整性	8	考虑不全（数据联动）	-
		考虑不全（网络差）	-
		考虑不全（多用户）	-
		考虑不全（版本兼容）	4
		边界测试	-
		功能完整性	4
需求变更	1	需求文档没明确	1
		需求变动	-
用户体验	6	交互处理	2
		样式	4
		兼容性	-
其他	-		-
外部系统	-		-



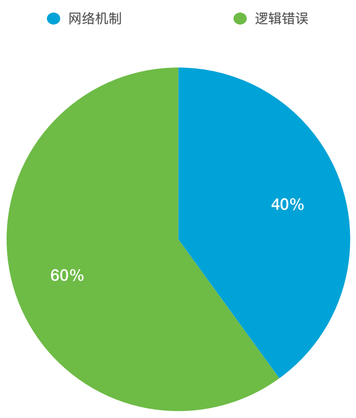
【性能问题】类型细分

BUG类型细分	数量
数据越界	0
数据类型	0
逻辑问题	0
版本兼容	0
未知原因	1



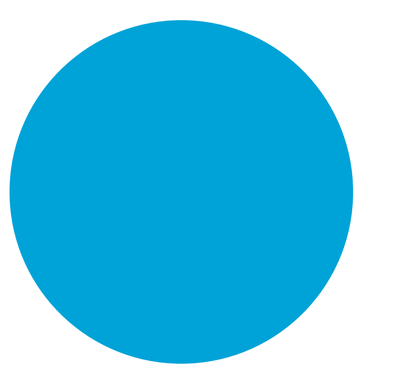
【代码逻辑】类型细分

BUG类型细分	数量
逻辑错误	6
网络机制	4



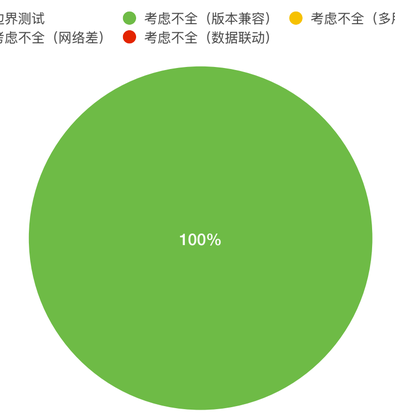
【接口相关】类型细分

BUG类型细分	数量
接口问题	-
环境问题	-
请求参数问题	-



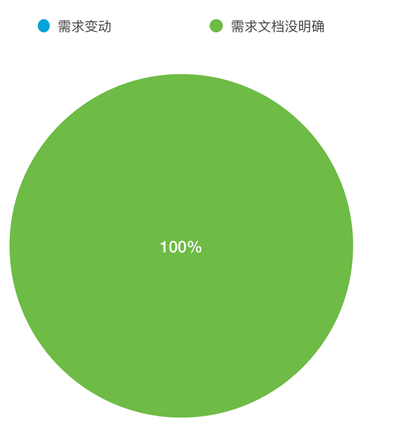
【完整性】类型细分

BUG类型细分	数量
考虑不全（数据联动）	-
考虑不全（网络差）	-
考虑不全（多用户）	-
考虑不全（版本兼容）	4
边界测试	-
功能完整性	4



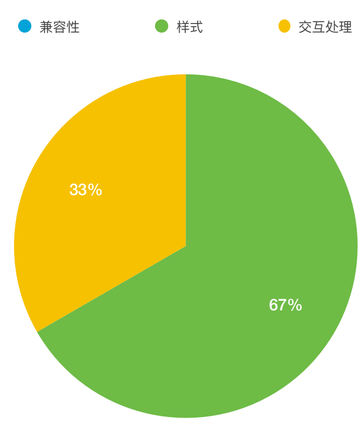
【需求变更】类型细分

BUG类型细分	数量
需求文档没明确	1
需求变动	-



【用户体验】类型细分

BUG类型细分	数量
交互处理	2
样式	4
兼容性	-



日均bug产生率

姓名	有效工作日	IOS 开发的BUG总数	BUG数/工作人日
潘思嘉	15	4	0.266666666666667
林俊炳	15	20	1.33333333333333
任鹏	12	0	0
总人日	42	24	0.571428571428571

bug激活情况

BUGID	激活次数	类型	所属人（以首个修复者为准）
95550	1	RN	林俊炳
95412	1	RN	任鹏
95521	1	vue	潘思嘉

上线一周crash情况

日期	CRASH数	CRASH率
2017-07-14	(android)16,(ios) 27	0.0021, 0.0005
2017-07-15	19, 30	0.002, 0.0006
2017-07-16	23, 25	0.002, 0.0005
2017-07-17	37, 20	0.002, 0.0004
2017-07-18	30, 19	0.0022, 0.0004
2017-07-19	21, 21	0.0014, 0.0004
2017-07-20	25, 19	0.0018, 0.0004

上线一周用户反馈及保障情况

日期	问题说明	结果
2017-20-00		

项目总结

姓名	总结
潘思嘉	已经沉淀下来的RN组件代码仍然在具体需求中发现不足，例如部分低端android 机动画不流畅：vue 的渲染功能也有缺陷，例如在ios 10.3 存在渲染图片失败的问题，导致实际需要的开发时间超出预期。 本次开发的前期，和惠璇等有丰富经营的同事的沟通不足，例如缺少考虑终端页也有复杂的网络请求情况，多花了时间在选择解决方案上，不够时间跑完测试用例。 小组成员的沟通方式还需改进，增强提问能力，了解合作同事的工作方式，提高默契。利用人员数量优势，通过约定分别在不同系统版本下开发，提前发现兼容问题。
林俊炳	聚超值是本人进入公司参与的第一个RN项目,实际是在原生基础上集成RN的工程实战,跟自己之前的纯RN开发经历还是有点区别。本次我负责的是部分列表、iOS端终端页的功能开发。这次开发参与下来,我最大的感觉还是快,然后也探到了不少坑,该模块的稳定性还得继续观察线上情况。模块的列表部分其实在两天之内就已经完成的了,总结了下,时间基本花费在以下几点: 1、网络请求方面：ipv6适配、httpdns处理 2、组件使用方面: 1）轮播组件在Android端存在手势冲突 2）列表组件在个别Android机型存在兼容问题 3）RN实现顶部的分类栏目切换动画 针对以上问题,目前在项目中做了如下处理: 1）网络请求采用的是第三方组件,为了适配ipv6,修改了组件的个别源码文件,使之支持公司iOS框架对ipv6的处理 2）在RN统一处理httpdns,发现status code为重定向时,进行转IP请求 3）按平台自定义轮播图 在做了这些技术尝试之后,我认为不管是目前的聚超值模块,还是后续的混合开发中,RN这块还有一些地方可按需进行改进优化的: 1、bundle的加载（减少首屏白屏时间） 2、丰富、复用及稳定的组件库（个人觉得较为重要的,根据公司业务自己维护更新一套） 3、符合公司业务情况的网络请求工具（ipv6、httpdns、缓存） 4、原生与RN混编的选型问题（APP中原生与RN的分别适合实现什么功能等）
任鹏	在本次项目中，由于同样采用了ReactNative来实现聚超值模块，又有iOS家居杂志的经验，于是绕开了许多坑。但是由于RN的某些组件在iOS平台和Android平台的实现方式上存在较大差异，于是bug也表现出平台差异性。例如初始化bundle的白屏问题，在Android平台上就特别明显，大概有1.2秒的时间，于是只能采取提前加载ReactRootView来优化此问题。又例如ScrollView的嵌套滑动，外层ScrollView会完全消耗掉滑动事件，导致内层ScrollView完全不能滑动。 总体上来讲，RN现在处于发展阶段，还有很多机制和组件是不完善的，像RN的事件分发机制和原生Android的事件分发机制还是有差别的，某些问题只能绕开，而不能真正解决。期待FaceBook能够早日出稳定版的RN，修复已知问题，提高开发效率。