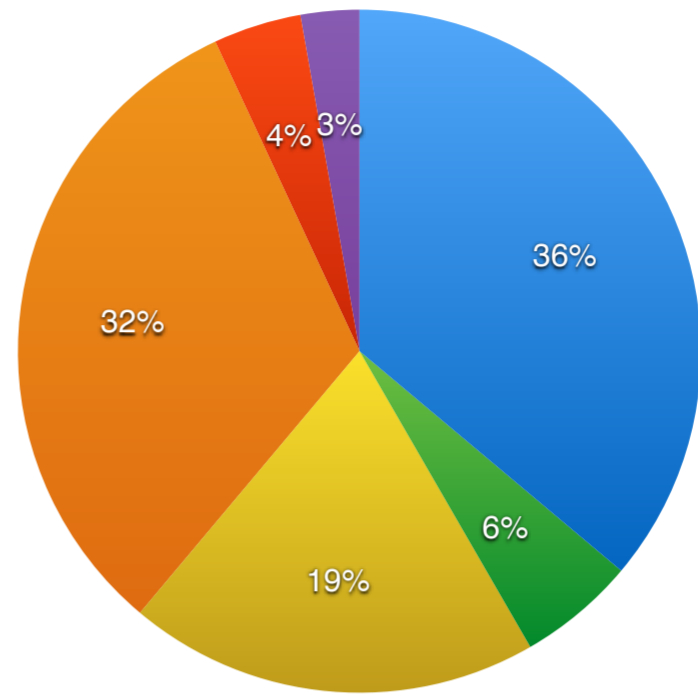


Bug类型

bug类型	数量 (72)	bug类型细分	数量
性能问题	3	数据越界	-
		数据类型	-
		逻辑问题	1
		系统版本兼容	2
		未知原因	-
代码逻辑	26	逻辑错误	26
		网络机制	-
接口相关	4	接口问题	4
		环境问题	-
		请求参数问题	-
完整性	14	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 (版本升级兼容)	-
		考虑不全 (项目兼容)	8
		边界测试	-
需求变更	2	需求文档没明确	-
		需求变动	2
用户体验	23	交互处理	5
		样式	18
		兼容性	-
其他	-		-
外部系统	-		-

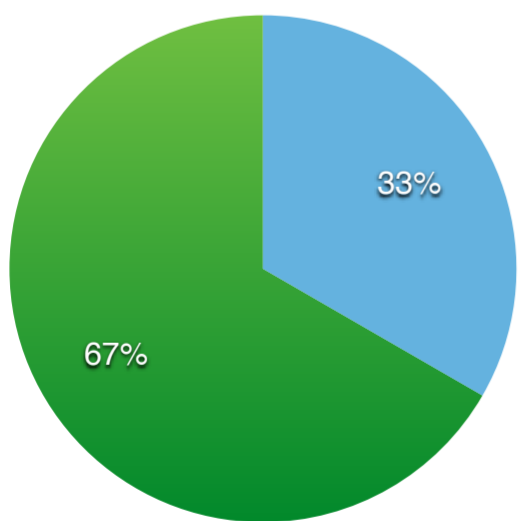
● 代码逻辑 ● 接口相关 ● 完整性 ● 用户体验 ● 性能问题 ● 需求变更



【性能问题】类型细分

bug类型	数量
逻辑问题	1
系统版本兼容	2

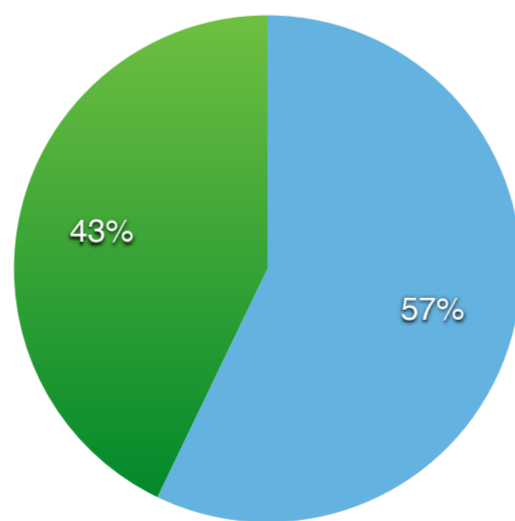
● 逻辑问题 ● 系统版本兼容



【完整性】类型细分

bug类型	数量
考虑不全 (项目兼容)	8
功能完整性	6

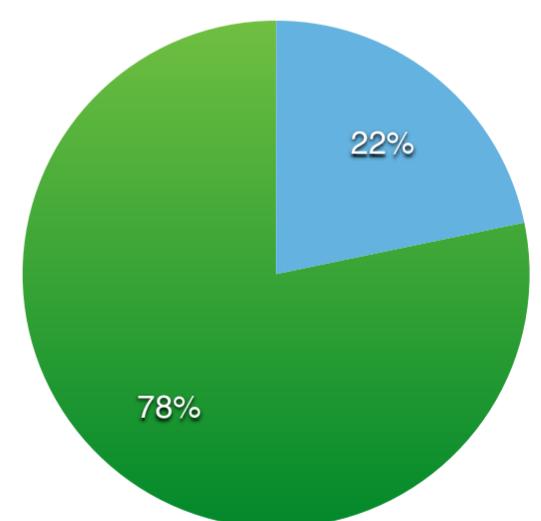
● 考虑不全 (项目兼容) ● 功能完整性



【用户体验】类型细分

bug类型	数量
交互处理	5
样式	18

● 交互处理 ● 样式



日均bug产生率

姓名	有效工作日	iOS开发的bug总数 (不包含需求变动)	bug数/工作人日
陈东贤	9	19	2.11
卢德智	9	7	0.78

姓名	有效工作日	iOS开发的bug总数（不包含需求变动）	bug数/工作人日
李冠淳	9	13	1.44
许伟杰	9	19	2.11
欧建强	12	11	0.92
总人日	48	69	1.44

#### Bug激活情况

BugID	激活次数 (0)	类型	所属人（以首个修复者为准）
87417	2	iOS	许伟杰
87669	2	iOS	李冠淳
87708	1	iOS	许伟杰
87795	1	iOS	许伟杰
87669	2	iOS	李冠淳
87888	1	iOS	许伟杰
88006	1	mrobot	傅兴方
88186	1	iOS	陈东贤

#### 激活率

姓名	bug激活数	bug总数	激活总次数/bug总数
总体	10	69	0.14

#### 上线一周日志

日期	crash数	crash率
2017-1-16	0	0
2017-1-17	1	0.0002
2017-1-18	1	0.0002
2017-1-19	1	0.0002
2017-1-20	0	0
2017-1-21	1	0.0002
2017-1-22	0	0

#### 项目总结

陈东贤	<p>车友5.0的主要任务是移植并同步资讯论坛的功能，涉及的地方非常广，有论坛广场、论坛列表页、帖子终端页及发帖、回帖整套功能等。而这次我负责的是论坛首页和列表部分。幸好项目前期工作建强做得非常足，避免了大部分的问题。</p> <p>我一开始对车友的逻辑架构并不熟悉，所以移植的开始没有考虑好各种兼容性的问题，同一套代码，在资讯和车友的执行的时机也会不一样，导致在测试阶段被QA提出各种奇怪的bg，很多问题本该在开发阶段就要发现的，也说明了自己本身的不足。</p> <p>5.0里面我修复的大多是样式兼容的bug，能快速解决。不过其中碰到两个很“有趣”bug，一个是索引条的样式问题 (<a href="http://bug.pconline.com.cn/bugfree/Bug.php?BugID=87788">http://bug.pconline.com.cn/bugfree/Bug.php?BugID=87788</a>)，这个索引条是从资讯上搬过来的，照理来说是没问题的，所一开始我考虑的是不是autolayout而导致的，但看了看层数结构就把这个原因排除掉，然后考虑是不是手势的问题，于是把整个项目翻了个遍，发现是和PCAppDelegate创建的一个手势冲突了，而这个手势在的触发方法就以前就去掉了，是一个“空”的手势。而第二个bug是帖子列表页的样式问题 (<a href="http://bug.pconline.com.cn/bugfree/Bug.php?BugID=88020">http://bug.pconline.com.cn/bugfree/Bug.php?BugID=88020</a>)，这个bug的原因在iOS10上通过textcore计算最后一行的字数和label实际渲染的字数不符，而由于我对textcore并不熟悉，所以花了很多时间去查找文档，虽然最后并不是用textcore去解决而是用了textkit,但也大大的加深了我对textcore的理解。</p>
卢德智	<p>这个版本主要是做资讯与车友论坛代码合并，方便维护论坛代码。主要与冠淳负责发帖和回帖部分代码同步。为了保持资讯与车友代码的同步，在资讯搬迁过来的如图片选择、发帖编辑页、回帖页等需要对车友原本所涉及到的部分做兼容，遇到的问题主要是：</p> <ol style="list-style-type: none"> <li>1、车友旧的业务逻辑不熟悉，许多细节未注意到；</li> <li>2、资讯中很多业务与车友的冲突；</li> <li>3、代码迁移遗漏等。</li> </ol>
李冠淳	<p>这次车友客户端5.0.0项目，资讯客户端论坛相关部分同车友客户端进行同步，我主要处理发帖部分及车友会相关列表页样式。由于之后这部分同步代码会共用，所以这次出现了很多资讯遗留的bug，特别是发帖部分，前前后后激活过4次。不过多亏这次棘手的Debug，对发帖整个流程理解得更加透彻了。不过，即使是棘手，激活的次数还是略多，希望之后能更加有效的解决bug，提前沟通好样式上的问题，更加有效进行开发。</p>

许伟杰	<p>这次做车友5.0.0版本，我主要负责的是”我的”模块里的UI调整、整合资讯的车友回复、我的帖子、我的草稿、我的收藏等几个页面、魔方计数。这期的bug很多，主要是因为很多代码是用资讯的，但是一些地方的实现方式、业务逻辑会存在一些差异，所以导致很多细节问题没有注意到，例如帖子回复-赞我的 有新消息时没有单独红点，是因为两个APP是用了不同的字段。另外在修改导航栏颜色时，漏了很多页面没有处理。原因之一是bug比较多，没有充足的时间去检查。另外自己解决问题的态度不够积极，常常没有及时地去找相关人员进行沟通。所以以后要认真及时地对待每一个问题！</p>
欧建强	<p>车友V5.0.0，主要需求是将汽车资讯的论坛模块移植到车友。时间有些紧，然后工作量也不少，主要是移植处理所牵涉的细节比较多，所以bug率普通偏高。以下是抽取并移植论坛代码的一些总结：</p> <p>一、抽离 抽离即是抽离出论坛相关的所有代码，作为公共代码提供给车友app和资讯app引用。 难点及对应处理： 1、没有很好的抽离方案。当时只想到2个方案： 1) 在资讯项目工程里操作：将论坛及相关功能抽出来放在独立的文件夹里，再引用然后删掉工程原有的重复文件。 优点是不要重新搭建环境，处理难度相对较低，缺点是很容易做漏。 2) 新建工程，搭好环境，再逐一将相关文件添加到该工程并修改和调试，直至编译通过为止。 优点是很大程度的减少做漏的情况，缺点是要重新搭建和配置开发环境。 尝试过方案1之后，最后采取方案2进行分离工作。 2、内容多，类文件引用关系复杂： 1) 论坛相关的内容很多，如论坛广场、论坛列表、发帖、回帖，草稿箱，我的帖子，帖子回复等等。 2) 以上页面会引用的“其他类”，而这些其他类也会引用别的类。 3) 所有类可能还用到宏和枚举。 3、类模块是否纳入公共模块的判断标准模糊 1) 只是论坛部分用到的，放共有。 2) 如果这些类是资讯特有的，如离线帖子数据，不放共有。 3) 只是资讯部分用到但论坛没用到却有引用的，删掉引用。 4) 大图浏览页面，考虑到车友别的地方也用到图片预览页，且使用的参数和规则可能还不一致，出于时间和风险的考虑，先不共用。 5) 所有影响编译的接口宏、消息宏、文件名宏、环境配置宏、枚举都暂时放在一个头文件里，后续再进行区别和处理。</p> <p>经过以上处理所得到的代码，严格来说还是未完整的，因为编译不会检查是否代码里所用到的一下资源文件，如Resources里的json文件，以及Images.xcassets里的图片文件。这个处理需要在后续的“兼容”步骤中完成。</p> <p>二、引用 将“抽离”步骤中所得到的共用文件引进车友工程里。 开始引用时，编译的结果必然是一大波Error。接下来的任务就是要处理这些Errors了。</p> <p>三、去重 去重即是去除车友工程中跟论坛公共代码重名的文件。 难点： 1、内容多，所以只能多人同时处理； 2、步骤繁琐： 1) 找 2) 有则记录（后续“兼容”会用到：找出所有引用到的代码并处理） 3) 记录后，再show in finder 4) 删除引用（选Move to Trash） 5) 被删类所在的文件夹没有别的文件时，删除该文件夹 6) continue...</p> <p>经过“去重”后，车友工程还是无法成功编译的，因为被删掉的类可能会有公共代码不包含的方法，此时就要逐个处理。这个处理只是粗略处理，目的是编译成功，否则其他组员无法开展开发工作了。</p> <p>四、兼容 兼容处理的目的就是在不影响资讯论坛原有业务功能的前提下，修改代码以移植论坛模块至车友app。 难点在于，由于兼容处理也是多人同时进行的，难免有代码冲突，或者有漏处理的情况。 1、逐一检查“去重”步骤中所记录的类逻辑，如果是论坛以外的模块用到的，则对比原有车友的代码逻辑，修改公共代码以兼容。 2、逐一检查“去重”步骤中所记录的类所引用的图片和json配置等资源文件，从资讯工程中copy过来，如果论坛以外的模块用到的，则检查相应位置的UI是否有影响，有则调整。 3、逐一检查“去重”步骤中所记录的类所用到的接口、通知、枚举、以及其他宏，如果论坛以外的模块用到且对功能有影响，有则调整。 4、逐一检查“去重”步骤中所记录的类所用到的协议，如果协议对应的跳转页面是资讯独有的功能，则加上宏编译条件控制，否则修改代码以兼容。</p> <p>以上步骤是项目完成后所归纳和总结的，当时处理前并没有很清晰的方案和步骤，这也是整个项目下来，bug偏多的一个原因吧。下次如果有类似的需求，可以作为一个参考方案。</p>