《sharing-bicycle》单元测试报告

目录

1 引言	3
1.1 编写目的	3
1.2 背景	3
1.3 定义	3
1.4 参考资料	3
2 测试概要	3
2.1 Bicycle 单元测试设计	3
2.2 Credit_record 单元测试设计	4
2.3 CreditActivity 单元测试设计	5
2.4 CreditAdapter 单元测试设计	5
2.5 LoginActivity 单元测试用例设计	6
2.6 NavigateActivity 单元测试用例设计	6
2.7 NetUtils 单元测试用例设计	6
2.8 Wallet_record 单元测试用例设计	7
2.9 WalletAdapter 单元测试用例设计	7
2.10 WalletActivity 单元测试用例设计	8
2.11 ActivityHelpContent 单元测试设计	8
2.12 HelpActivity 单元测试设计	
2.13 ChartActivity 单元测试设计	
2.14 ChartAdapter 单元测试设计	10
2.15 ChartFragment 单元测试设计	10
2.16 ConsumeFragment 单元测试设计	
2.17 DistanceUtils 单元测试设计	11
2.18 GetCreditFragment 单元测试设计	
2.19 NetUtils 单元测试设计	11
2.20 PreorderHelp 单元测试设计	12
2.21 PreorderActivity 单元测试设计	13
2.22 MainActivity 单元测试设计	13
3 测试结果及发现	15
3.1 Bicycle 单元测试结果	
3.2 Credit_record 单元测试结果	
3.3 CreditActivity 单元测试结果	
3.4 CreditAdapter 单元测试结果	
3.5 LoginActivity 单元测试结果	
3.6 NavigateActivity 单元测试结果	19
3.7 NetUtils 单元测试结果	
3.8 Wallet_record 单元测试结果	21
3.9 WalletAdapter 单元测试结果	
3.10 WalletActivity 单元测试结果	
3.11 ActivityHelpContent 单元测试结果	22
3.12 HelpActivity 单元测试结果	23
3.13 ChartActivity 单元测试结果	25
3.14 ChartAdapter 单元测试结果	25

3.15 ChartFragment 单元测试结果	25
3.16 ConsumeFragment 单元测试结果	26
3.17 DistanceUtils 单元测试结果	26
3.18 GetCreditFragment 单元测试结果	27
3.19 NetUtils 单元测试结果	27
3.20 PreorderHelp 单元测试结果	28
3.21 PreorderActivity 单元测试结果	29
3.22 MainActivity 单元测试结果	30
4 对软件功能的结论	32
4.1 功能	32
4.2 界面	33
4.3 能力	38
5 分析摘要	39
5.1 缺陷和限制	39
5.2 评价	39
6 测试资源消耗	40

1引言

1.1 编写目的

根据测试计划书,对软件进行测试,详细记录测试过程以及测试结果,以对测试结果进行分析,对软件的质量进行评价,为软件设计人员提供 BUG 依据,故编写此测试分析报告。

1.2 背景

随着共享单车的大规模普及,人们在享受共享单车带来的方便的同时,也对共享单车件存在的许多问题有着不满。如今的共享单车软件在很多方面仍然存在着问题,如:由于二维码受到破坏而无法使用单车,用户由于无法预约车辆而导致在找车上浪费大量时间,等等;本软件针对当下共享单车存在的问题做出改进,提升用户体验。让共享单车的使用者能够在寻找车辆时节省时间,以全新的解锁方式使单车解锁更加方便,让共享单车的使用者能够享受到骑行的方便,同时提高共享单车的使用率。

1.3 定义

Android: 是一种基于 Linux 的自由及开放源代码的操作系统,主要使用于移动设备。 共享单车: 是指企业在校园、地铁站点、公交站点、居民区、商业区、公共服务区等提供自 行车单车共享服务,是一种分时租赁模式。

1.4 参考资料

- 1. 《软件工程——实践者的研究方法》 Roger S. Pressman, Bruce R. Maxim 编著, 机械工业出版社, 2015
 - 2. 《sharing-bicycle 软件需求说明书》.2017 年 4 月 19 日 .黎文彬、宋羽珩
 - 3. 《sharing-bicycle 共享单车概要设计》.2017年5月9日. 黎文彬、宋羽珩

2测试概要

2.1 Bicycle 单元测试设计

测试 ID 测	则试名称	前置条件	输入数据	期待输	优先级
---------	------	------	------	-----	-----

				出	
Bic-getBi	Id_get 方法	构造 Bicycle 对	{bike_id:1}	{1}	Low
ke_id		象			
Bic_setBi	Id_set 方法	构造 Bicycle 对	{bike_id:5}	{5}(get	Low
ke_id		象		方法测	
				试)	
Bic_getLo	Location_ge	构造 Bicycle 对	{location:(104.001904,	{(104.0	Low
cation	t方法	象	30.560536)}	01904,3	
				0.56053	
				6)}	
Bic_setLo	Location_se	构造 Bicycle 对	{location:(103.99655,3	{(103.9	Low
cation	t方法	象	0.565947)}	9655,30.	
				565947)	
				}	
				(get 方	
				法测试)	
Bic_isIn_	Inuse_get方	构造 Bicycle 对	{in_use(boolean):false}	{false}	Low
use	法	象			
Bic_setIn_	Inuse_set 方	构造 Bicycle 对	{in_use(boolean):true}	{true}(g	Low
use	法	象		et 方法	
		17 No.		测试)	_
Bic_isBre	Isbreakdow	构造 Bicycle 对	{isbreakdown(boolean):	{true}	Low
akdown	n_get 方法	象	true}	(0.4.)	_
Bic_setBr	Breakdown	构造 Bicycle 对	{ isbreakdown(boolean)	{false}	Low
eakdown	_set 方法	象	:false }	(get 方	
D' 11	T.1 1	护 P 1 31	(1 1 1 1) 6	法测试)	T
Bic_Id_pr	Idpreorder_	构造 Bicycle 对	{id_preorder(boolean):f	{false}	Low
eorder	get 方法	象 物选 P:1 对	alse}	(4	T
Bic_setId_	Idpreorder_	构造 Bicycle 对	{id_preorder(boolean):t	{true}(g	Low
preorder	set 方法	象	rue}	et 方法	
Dia catDa	Description	机类 Diavide 对	(decomination (Stair a), #	测试)	Low
Bic_getDe	Description	构造 Bicycle 对 象	{description(String): 车 辆故障}	{车辆故	Low
scription Ric set Do	_get		-	[左股沿	Low
Bic_setDe	Description	构造 Bicycle 对 象	{description(String): 车 胎没气}	{车胎没	LOW
scription	_set		加仪()	气}	

2.2 Credit_record 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Cre_geta	getCredit_a	构造	{credit_amount(String):	{100}	Low
mount	mount()	credit_record 对	100}		
		象			

Cre_setam	setCredit_a	构造	{credit_amount(String):	{95}	Low
ount	mount()	credit_record 对	95}		
		象			
Cre_getda	getDate_tim	构造	{date_time(String):2017	{2017-5-20	Low
te	e()	credit_record 对	-5-20}	}	
		象			
Cre_setdat	setDate_tim	构造	{date_time(String):2017	{2017-5-26	Low
e	e()	credit_record 对	-5-26}	}	
		象			
Cre_getde	getDescripti	构造	{description(String): 骑	{骑行奖励}	Low
scription	on()	credit_record 对	行奖励}		
		象			
Cre_setDe	setDescripti	构造	{description(String): 报	{报修奖励}	Low
scription	on()	credit_record 对	修奖励}		
		象			

2.3 CreditActivity 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Cred_onR	列表刷新方	onCreate() 方法	{下拉列表}	返回列表数	Medi
efresh	法	执行		据	um
Cred_clic	返回键监听	onCreate() 方法	{点击返回键}	返回主界面	Medi
k	方法	执行			um
Cred_onb	返回按钮监	onCreate() 方法	{点击返回按钮}	返回主界面	Medi
ack	听	执行			um

2.4 CreditAdapter 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Credit_get	getCount()	List <credit_recor< td=""><td>无</td><td>{size(int)}</td><td>Medi</td></credit_recor<>	无	{size(int)}	Medi
Count		d>构造完成			um
Credit_get	getItem()	List <credit_recor< td=""><td>无</td><td>{返回点击</td><td>Medi</td></credit_recor<>	无	{返回点击	Medi
Item		d>构造完成		的列表项}	um
Credit_get	getItemId()	List <credit_recor< td=""><td>无</td><td>{position(in</td><td>Medi</td></credit_recor<>	无	{position(in	Medi
ItemId		d>构造完成		t)}	um
Credit_set	setText1()	List <credit_recor< td=""><td>{date:2017-5-21}</td><td>{2017-5-21</td><td>Medi</td></credit_recor<>	{date:2017-5-21}	{2017-5-21	Medi
Text1		d>构造完成		}(get 方法	um
				测试)	
Credit_set	setText2()	List <credit_recor< td=""><td>{"积分变化"}</td><td>{"积分变</td><td>Medi</td></credit_recor<>	{"积分变化"}	{"积分变	Medi

Text2		d>构造完成		化"}(get 方	um
				法测试)	
Credit_set	setText3()	List <credit_recor< td=""><td>{"备注"}</td><td>{ " 备</td><td>Medi</td></credit_recor<>	{"备注"}	{ " 备	Medi
Text3		d>构造完成		注"}(get 方	um
				法测试)	
Credit_get	获取列表试	List <credit_recor< td=""><td>{环境上下文: context}</td><td>{返回列表</td><td>Medi</td></credit_recor<>	{环境上下文: context}	{返回列表	Medi
View	图	d>构造完成		试图}	um

2.5 LoginActivity 单元测试用例设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Login_que	查询用户积	用户登录后	{user_id(Stirng):13577	{balance(do	Medi
ry	分方法		776666}	uble),credit(um
				double)}	
Login_clic	登录按钮方	无	{id(String) :	{flag:true}	High
k	法		13577776666,passwor		
			d(String): root123}		
Login_reg	注册界面跳	无	点击注册按钮	跳转到注册	Medi
ister	转			界面	um
Login_bac	返回键监听	无	点击返回键	返回主界面	Medi
k					um

2.6 NavigateActivity 单元测试用例设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Navi_onG	搜索目的地	地图视图	{query(String):成都}	显示搜索列	Medi
etInput	方法			表	um
Navi_onIt	列表项点击	地图视图	{Itemname(String): 四	显示当前位	Medi
mclick	方法		川大学(望江校区) }	置到目的地	um
				的路径	
Navi_back	返回监听方	无	无	跳转	Medi
	法			Mainactivit	um
				y	

2.7 NetUtils 单元测试用例设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优 先 级
Net_get	GET 请求 方法	无	{url(String): "http://123.206.80.243:	{credit(Strin g):100}	High

			8080/sharing_bicycle/		
			wallet_credit.do;parms		
			userid(String):1357777		
			6666}		
Net_post	Post 请求	无	{url(String):	{success(Str	High
	方法		http://123.206.80.243:8	ing)}	
			080/sharing_bicycle/fi		
			nish_preorder.do;parm		
			suserid(String):135777		
			76666 ,		
			bike_id(String):1}		

2.8 Wallet_record 单元测试用例设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Wall_geta	getAmount(构 造	{amount(String):100.0	100.00	Low
mount)	wallet_record 对	0}		
		象			
Wall_seta	setAmount(构 造	{amount(String):99.5}	99.5	Low
mount)	wallet_record 对			
		象			
Wall_getd	getDate_tim	构 造	{date_time(String):201	{2017-5-25	Low
ate	e()	wallet_record 对	7-5-25}	}	
		象			
Wall_setd	setDate_tim	构 造	{date_time(String):201	{2017-5-27	Low
ate	e()	wallet_record 对	7-5-27}	}	
		象			

2.9 WalletAdapter 单元测试用例设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
Wallet_get	getCount()	List <wallet_reco< td=""><td>无</td><td>{size(int)}</td><td>Medi</td></wallet_reco<>	无	{size(int)}	Medi
Count		rd>构造完成			um
Wallet_get	getItem()	List <wallet_reco< td=""><td>无</td><td>{返回点击</td><td>Medi</td></wallet_reco<>	无	{返回点击	Medi
Item		rd>构造完成		的列表项}	um
Wallet_get	getItemId()	List <wallet_reco< td=""><td>无</td><td>{position(in</td><td>Medi</td></wallet_reco<>	无	{position(in	Medi
ItemId		rd>构造完成		t)}	um
Wallet_set	setText1()	List <wallet_reco< td=""><td>{date:2017-5-21}</td><td>{2017-5-21</td><td>Medi</td></wallet_reco<>	{date:2017-5-21}	{2017-5-21	Medi
Text1		rd>构造完成		}(get 方法	um
				测试)	

Wallet_set	setText2()	List <wallet_reco< th=""><th>{"消费"}</th><th>{ "消</th><th>Medi</th></wallet_reco<>	{"消费"}	{ "消	Medi
Text2		rd>构造完成		费"}(get 方	um
				法测试)	
Wallet_set	setText3()	List <wallet_reco< td=""><td>{"消费时间"}</td><td>{"消费时</td><td>Medi</td></wallet_reco<>	{"消费时间"}	{"消费时	Medi
Text3		rd>构造完成		间"}(get 方	um
				法测试)	
Wallet_get	获取列表试	List <wallet_reco< td=""><td>{环境上下文: context}</td><td>{返回列表</td><td>Medi</td></wallet_reco<>	{环境上下文: context}	{返回列表	Medi
View	图	rd>构造完成		试图}	um

2.10 WalletActivity 单元测试用例设计

测试 ID	测试名称	前置条件	f置条件 输入数据		优先
					级
Cred_onR	列表刷新方	onCreate() 方法	onCreate() 方法 {user_id(Stirng):13377		Medi
efresh	法	执行	776666}	ing):100.00	um
				}	
Cred_clic	返回键监听	onCreate() 方法	{点击返回键}	返回主界面	Medi
k	方法	执行			um
Cred_onb	返回按钮监	onCreate() 方法	{点击返回按钮}	返回主界面	Medi
ack	听	执行			um

2.11 ActivityHelpContent 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优 先 级
ActivityHel pContent	对应的请求是否能显示相应的帮助信息	无	{int type:1}	(向左划打 开侧滑"登录/注册", 进入登录/车"输入 面,在"输入。	级 low
				经注册在"的账箱"的"输入"的"输入"的"通过"的"从后"。 是一个,是一个,是一个,是一个,是一个,是一个,是一个,是一个,是一个,是一个,	

ActivityHel	返回键监听	无	{按下返回键 1}	{执行返回	low
pContent_b	方法			主界面动	
ack				作}	

2.12 HelpActivity 单元测试设计

测试 ID	测试名称	前置条	输入数据	期待输出	优先级
		件			
HelpActivity_back	返回键监听方	无	{按下返	{执行返	low
	法		回键 1}	回主界面	
				动作}	
HelpActivity_list	列表点击事件	列表已	{点击列	{执行跳	Medium
		经加载	表 第 一	转到列表	
		所有的	项}	第一项所	
		帮助信		对应的内	
		息变量		容动作}	
HelpActivity_list_title	列表标题与内	列表已	{点击列	{执行跳	Medium
	容的对应	经加载	表 第 一	转到列表	
		所有的	项}	第一项所	
		帮助信		对应的内	
		息标题		容,内容	
		项		标题与列	
				表第一项	
				标题相同	
				动作}	
HelpActivity_list_gettitle	测 试	列表已	无	{ List(含	high
	getTutorialTitle	经加载		有帮助内	
	方法	所有的		容标题)}	
		帮助信			
		息标题			
		项			

2.13 ChartActivity 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
ChartActivity_back	返回键监	无	{按下返回	{执行返回	low
	听方法		键}	主界面动	
				作}	

2.14 ChartAdapter 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
ChartAdapter1	getItem 方法	所 有 的	{int position:	{ fragment	High
		fragment	1}	(第二	
		已经初始		个) }	
		化完毕			
ChartAdapter2	getCount 方法	所 有 的	{无}	{int : 5}	Low
		fragment			
		已经初始			
		化完毕			
ChartAdapter3	getPageTitle	所 有 的	{int position:	{String: 消	High
	方法	fragment	1}	费金额}	
		已经初始			
		化完毕			

2.15 ChartFragment 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
ChartFragment1	initControls	无	{View view:	{界面显示	High
	方法		当前页面的	所有的列	
			view}	表的标题}	

2.16 ConsumeFragment 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
ConsumeFragment1	queryData 方法	无	{无}	{Vector:	High
				data_raw}	
ConsumeFragment2	generateDefaultData	已经从服	{无}	{ 将	High
	方法	务器获得		data_raw	
		所需数据		的数据截	
		并装载在		取最近 7	
		data_raw		条加载在	
		中		图表上}	
ConsumeFragment3	resetViewport 方法	无	{无}	{设置了	Low
				坐标轴信	
				息 的 图	
				标}	
ConsumeFragment4	onValueSelected 方	所有的数	{ int	{ Snackbar	Medium
	法	据已经装	columnIndex: 1,	提示点击	
		载在图上	int	的 value	
			subcolumnIndex:1,	值}	

	SubcolumnValue	
	value}	

2.17 DistanceUtils 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
DistanceUtils1	getDistanceOfMeter	已经获得	{double lat1:	{double:	High
	方法	特定目标	30.557498,	1037.9}	
		点的经纬	double lng1:		
		度	103.998449,		
			double lat2:		
			30.563381,		
			double lng2:		
			104.006849}		

2.18 GetCreditFragment 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先级
GetCreditFragment1	queryData 方法	无	{空}	{int[] :	High
				each_type	
				(包含从	
				服务器请	
				求会的所	
				有数据封	
				装得到的	
				数组)}	
GetCreditFragment2	Reset 方法	无	{空}	{重置之后	Low
				的图表}	
GetCreditFragment3	generateData 方	已经从服务	{空}	{将数据加	High
	法	器获取所需		载之后的	
		数据并且加		图表}	
		载 在			
		each_type			
		上			

2.19 NetUtils 单元测试设计

测试 ID	测试名称	前置条件	前置条件 输入数据 其		优先
					级
NetUtils1	getRequest	获取参数	{final String url:	{执行动	High
	方法	param	http://123.206.80.243:808	作:新开	

			0/sharing_bicycle/bike_f	一个线程	
			ull, final	向服务器	
			Map <string,string></string,string>	发送查询	
			params, final Handler	车辆信息	
			handler}	请求}	
NetUtils2	postRequest	获 取	{final String url :	{执行动	High
	方法	String	http://123.206.80.243:808	作: 向服	
		URL和参	0/sharing_bicycle/finish_	务器发送	
		数 param	preorder.do, final	一个完成	
			List <basicnamevaluepai< th=""><th>预约请求</th><th></th></basicnamevaluepai<>	预约请求	
			r> params, final Handler	并获取返	
			handler}	回的结果	
				(含有	
				true 或者	
				false) }	

2.20 PreorderHelp 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先
					级
PreorderHelp1	PreorderHel	无	{无}	{ Preorde	High
	p方法			rHelp 对	
				象:	
				time_inte	
				rval_help	
				=-1,time	
				Runnable	
				=null }	
PreorderHelp2	getPreorder	Preorder	{final String url :	{执行动	High
	Help 方法	Help 对象	http://123.206.80.243:808	作: 向服	
		已经被初	0/sharing_bicycle/finish_	务器发送	
		始化	preorder.do, final	一个完成	
			List <basicnamevaluepai< td=""><td>预约请求</td><td></td></basicnamevaluepai<>	预约请求	
			r> params, final Handler	并获取返	
			handler}	回的结果	
				(含有	
				true 或者	
				false) }	
PreorderHelp3	getTime_int	Preorder	{空}	{int:对象	High
	erval_help	Help对象		的属性	

	方法	已经被初		time_inte	
		始化		rval_help	
				}	
PreorderHelp4	setTime_inte	无	{int:32}	{无}	High
	rval_help 方				
	法				
PreorderHelp5	getPreorder_	Preorder	{空}	{String:	High
	bike_id 方法	Help 对象		对象的属	
		已经被初		性:	
		始化		preorder_	
				bike_id}	
PreorderHelp6	setPreorder_	无	{String	{无}	high
	bike_id 方法		preorder_bike_id:"1"}		
PreorderHelp7	getTimeRun	Preorder	{无}	{String:	High
	nable 方法	Help 对象		对象的属	
		已经被初		性:	
		始化		timeRunn	
				able}	
PreorderHelp8	setTimeRun	无	{Runnable	{无}	High
	nable 方法		timeRunnable: runnable}		

2.21 PreorderActivity 单元测试设计

测试 ID	测试名称	前置条件	输入数据	期待输出	优先 级
PreorderActivity1	onBackPress	无	{返回键事件}	{执行动	Low
	ed 方法			作:跳转	
				到 主 界	
				面}	
PreorderActivity2	onTimeSet	无	{TimePicker view: 当前	{将显示	High
	() 方法 1		view, int hourOfDay: 所	的时间置	
			选择时间的小时数, int	为所选择	
			minute: 所选择时间的分	的时间}	
			钟数 }		
PreorderActivity3	onClick ()	已经选择	无	{执行动	High
	方 法	合理的预		作:向服	
	(confirmB	约时长		务器发送	
	utton)			请求}	

2.22 MainActivity 单元测试设计

测试 ID 测试名称	前置条件 输入数据	期待输出	优先
------------	-----------	------	----

					级
MainActivity1	onSaveInsta nceState 方 法	User 对象 不为空, timeremai nning 已 经初始化	{Bundle outState: 系统预设, PersistableBundle outPersistentState: 系统预设}	{无}	Low
MainActivity2	onResume 方法	User 对象 不为空, timeremai nning 和 mainPreH elp.getTi me_interv al_help都 为-1	{无}	{无}	High
MainActivity3	onResume 方法	User 对象 不为空, timeremai nning 或 mainPreH elp.getTi me_interv al_help 不 为-1	{无}	{执行动作:显示倒计时图标并倒计时	High
MainActivity4	queryBicycl eLoc 方法	无	{无}	{Vector bicycles}	High
MainActivity5	count_time 方法	预约已经 成功并且 未到达预 约时长	{int interval: 剩余的预约时间秒数}	{执行动 作:显示 倒计时}	High
MainActivity6	findBicycle ByID 方法	无	{String initID: 1}	{Bicycle: 1 号车辆 对象}	Med ium
MainActivity7	Scroll 方法	背景图像 view 已经 初始化	{无}	{执行动作:背景图像规律移动}	Low
MainActivity8	onBackPress ed 方法	无	{返回键事件}	{执行动 作: 侧滑 菜 单 关 闭}	Low
MainActivity9	registerWige t 方法	无	{无}	{初始化 控件}	High
MainActivity10	queryUserW	无	{String ID :	{double[]	High

	alletAndCre		12345678911}	: 账 号	
	dit 方法			为	
				123456	
				78911	
				对应的	
				余额和	
				积分数	
				组}	
MainActivity11	proccessAcc	用户已经	{String account :	{String:	Low
	ount 方法	登录	12345678911}	123***	
				*8911}	
MainActivity12	addMarkerT	已经获得	{空}	{将车辆	High
	oMap 方法	车辆位置		表示在	
		信息		地 图	
				上}	

3 测试结果及发现

3.1 Bicycle 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Bic-getB	Id_get 方	构 造	{bike_id	{1}	Low	{1}	宋羽珩	P
ike_id	法	Bicycle 对	:1}					
		象						
Bic_setB	Id_set 方	构 造	{bike_id	{5}(get	Low	{5}	宋羽珩	P
ike_id	法	Bicycle 对	:5}	方法测				
		象		试)				
Bic_get	Location_	构 造	{locatio	{(104.0	Low	{(104.0019	宋羽珩	P
Location	get 方法	Bicycle 对	n:(104.0	01904,3		04,30.56053		
		象	01904,3	0.56053		6)}		
			0.56053	6)}				
			6)}					
Bic_setL	Location_	构 造	{locatio	{(103.9	Low	{(103.9965	宋羽珩	P
ocation	set 方法	Bicycle 对	n:(103.9	9655,30.		5,30.565947		
		象	9655,30.	565947))}		

			565947)	}				
			}	(get方法				
				测试)				
Bic_isIn	Inuse_get	构 造	{in_use({false}	Low	{false}	宋羽珩	P
_use	方法	Bicycle 对	boolean)					
		象	:false}					
Bic_setI	Inuse_set	构 造	{in_use({true}(g	Low	{true}	宋羽珩	P
n_use	方法	Bicycle 对	boolean)	et 方法				
		象	:true}	测试)				
Bic_isBr	Isbreakdo	构 造	{isbreak	{true}	Low	{true}	宋羽珩	P
eakdown	wn_get 方	Bicycle 对	down(bo					
	法	象	olean):tr					
			ue}					
Bic_setB	Breakdow	构 造	{ isbreak	{false}	Low	{false}	宋羽珩	P
reakdow	n_set 方法	Bicycle 对	down(bo	(get方法				
n		象	olean):fa	测试)				
			lse }					
Bic_Id_	Idpreorder	构 造	{id_preo	{false}	Low	{false}	宋羽珩	P
preorder	_get 方法	Bicycle 对	rder(boo					
		象	lean):fal					
			se}					
Bic_setI	Idpreorder	构 造	{id_preo	{true}(g	Low	{true}	宋羽珩	P
d_preord	_set 方法	Bicycle 对	rder(boo	et 方法				
er		象	lean):tru	测试)				
			e}					
Bic_get	Descriptio	构 造	{descript	{车辆故	Low	{车辆故障}	宋羽珩	P
Descripti	n_get	Bicycle 对	ion(Strin	障}				
on		象	g): 车辆					
			故障}					
Bic_set	Descriptio	构 造	{descript	{车胎没	Low	{车胎没气}	宋羽珩	P
Descripti	n_set	Bicycle 对	ion(Strin	气}				
on		象	g): 车 胎					
			没气}					

3.2 Credit_record 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Cre_geta	getCredit_	构造	{credit_	{100}	Low	{100}	宋羽珩	P
mount	amount()	credit_rec	amount(
		ord 对象	String):					

			100}					
Cre_seta	setCredit_	构造	{credit_	{95}	Low	{95}	宋羽珩	P
mount	amount()	credit_rec	amount(
		ord 对象	String):					
			95}					
Cre_getd	getDate_ti	构造	{date_ti	{2017-5-	Low	{2017-5-20	宋羽珩	P
ate	me()	credit_rec	me(Strin	20}		}		
		ord 对象	g):2017-					
			5-20}					
Cre_setd	setDate_ti	构造	{date_ti	{2017-5-	Low	{2017-5-26	宋羽珩	P
ate	me()	credit_rec	me(Strin	26}		}		
		ord 对象	g):2017-					
			5-26}					
Cre_getd	getDescrip	构造	{descript	{骑行奖	Low	{骑行奖励}	宋羽珩	P
escriptio	tion()	credit_rec	ion(Strin	励}				
n		ord 对象	g): 骑 行					
			奖励}					
Cre_set	setDescrip	构造	{descript	{报修奖	Low	{报修奖励}	宋羽珩	P
Descripti	tion()	credit_rec	ion(Strin	励}				
on		ord 对象	g): 报修					
			奖励}					

3.3 CreditActivity 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Cred_on	列表刷新	onCreate()	{下拉列	返回列	Mediu	列表数据	宋羽珩	P
Refresh	方法	方法执行	表}	表数据	m			
Cred_cli	返回键监	onCreate()	{点击返	返回主	Mediu	成功跳转	宋羽珩	P
ck	听方法	方法执行	回键}	界面	m			
Cred_on	返回按钮	onCreate()	{点击返	返回主	Mediu	成功跳转	宋羽珩	P
back	监听	方法执行	回接钮}	界面	m			

3.4 CreditAdapter 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P
			据	出	先	际	行	/
					级	输	人	F
						出		
Credit_g	getCount()	List <credi< td=""><td>无</td><td>{size(int</td><td>Mediu</td><td>{size(int):6}</td><td>宋羽珩</td><td>P</td></credi<>	无	{size(int	Mediu	{size(int):6}	宋羽珩	P
etCount		t_record>)}	m			
		构造完成						
Credit_g	getItem()	List <credi< td=""><td>无</td><td>{返回点</td><td>Mediu</td><td>{返回点击</td><td>宋羽珩</td><td>P</td></credi<>	无	{返回点	Mediu	{返回点击	宋羽珩	P
etItem		t_record>		击的列	m	的列表项}		
		构造完成		表项}				
Credit_g	getItemId(List <credi< td=""><td>无</td><td>{position</td><td>Mediu</td><td>{position(in</td><td>宋羽珩</td><td>P</td></credi<>	无	{position	Mediu	{position(in	宋羽珩	P
etItemId)	t_record>		(int)}	m	t):5}		
		构造完成						
Credit_s	setText1()	List <credi< td=""><td>{date:20</td><td>{2017-5-</td><td>Mediu</td><td>{2017-5-21</td><td>宋羽珩</td><td>P</td></credi<>	{date:20	{2017-5-	Mediu	{2017-5-21	宋羽珩	P
etText1		t_record>	17-5-21}	21}(get	m	}		
		构造完成		方法测				
				试)				
Credit_s	setText2()	List <credi< td=""><td>{"积分</td><td>{"积分</td><td>Mediu</td><td>{"积分变</td><td>宋羽珩</td><td>P</td></credi<>	{"积分	{"积分	Mediu	{"积分变	宋羽珩	P
etText2		t_record>	变化"}	变	m	化"}		
		构造完成		化"}(get				
				方法测				
				试)				
Credit_s	setText3()	List <credi< td=""><td>{ " 备</td><td>{ " 备</td><td>Mediu</td><td>{"备注"}</td><td>宋羽珩</td><td>P</td></credi<>	{ " 备	{ " 备	Mediu	{"备注"}	宋羽珩	P
etText3		t_record>	注"}	注"}(get	m			
		构造完成		方法测				
				试)				
Credit_g	获取列表	List <credi< td=""><td>{环境上</td><td>{返回列</td><td>Mediu</td><td>{返回列表</td><td>宋羽珩</td><td>P</td></credi<>	{环境上	{返回列	Mediu	{返回列表	宋羽珩	P
etView	试图	t_record>	下文:	表试图}	m	试图}		
		构造完成	context}					

3.5 LoginActivity 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Login_q	查询用户	用户登录	{user_i	{balance	Mediu	{balance(do	宋羽珩	P
Login_q uery	查询用户 积分方法	用户登录 后	{user_i d(Stirng	{balance (double),	Mediu m	{balance(do uble) :	宋羽珩	P
			` _	`		,	宋羽珩	P

			}			100}		
Login_cl	登录按钮	无	{id(Stri	{flag:tru	High	{flag:true}	宋羽珩	P
ick	方法		ng) :	e}				
			135777					
			76666,p					
			assword					
			(String)					
			:					
			root123					
			}					
Login_re	注册界面	无	点击注	跳转到	Mediu	成功跳转	宋羽珩	P
gister	跳转		册按钮	注册界	m			
				面				
Login_b	返回键监	无	点击返	返回主	Mediu	成功跳转	宋羽珩	P
ack	听		回键	界面	m			

3.6 NavigateActivity 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Navi_on	搜索目的	地图视图	{query(显示搜	Mediu	{result[]: 列	宋羽珩	P
GetInput	地方法		String):	索列表	m	表(含北		
			成都}			京)}		
Navi_on	列表项点	地图视图	{Itemna	显示当	Mediu	显示当前位	宋羽珩	P
Itmclick	击方法		me(Stri	前位置	m	置到四川大		
			ng): 四	到目的		学望江校区		
			川大学	地的路		的路径		
			(望江	径				
			校					
			区) }					
Navi_ba	返回监听	无	无	跳转	Mediu	成功跳转	宋羽珩	P
ck	方法			Mainacti	m			
				vity				

3.7 NetUtils 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		

Net_get	GET 请求	无	{url(Stri	{credit(S	High	{credit(Strin	宋羽珩	P
1101_501	方法	/ 1	ng):	tring):10	111511	g):100}	NI2.445I1	•
	1/14		"http://1	0}		5).100]		
			23.206.	٠,				
			80.243:					
			8080/sh					
			aring_bi					
			cycle/w					
			allet_cr					
			edit.do;					
			parmsus					
			erid(Stri					
			ng):135					
			777766					
			66}					
Net_post	Post 请求	无	{url(Stri	{success	High	有缺陷,参	宋羽珩	F
	方法		ng):	(String)}		数不完整,		
			http://1			无返回数据		
			23.206.					
			80.243:					
			8080/sh					
			aring_bi					
			cycle/fi					
			nish_pr					
			eorder.d					
			o;parms					
			userid(S					
			tring):1					
			357777					
			6666,					
			bike_id(
			String):					
			1}					

3.8 Wallet_record 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Wall_get	getAmoun	构 造	{amoun	100.00	Low	100.00	宋羽珩	P
amount	t()	wallet_re	t(String					
		cord 对象):100.00					
			}					
Wall_set	setAmoun	构 造	{amoun	99.5	Low	99.5	宋羽珩	P
amount	t()	wallet_re	t(String					
		cord 对象):99.5}					
Wall_get	getDate_ti	构 造	{date_ti	{2017-5-	Low	{2017-5-25	宋羽珩	P
date	me()	wallet_re	me(Stri	25}		}		
		cord 对象	ng):201					
			7-5-25}					
Wall_set	setDate_ti	构 造	{date_ti	{2017-5-	Low	{2017-5-27	宋羽珩	P
date	me()	wallet_re	me(Stri	27}		}		
		cord 对象	ng):201					
			7-5-27}					

3.9 WalletAdapter 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Wallet_g	getCount()	List <walle< td=""><td>无</td><td>{size(int</td><td>Mediu</td><td>{size(int):</td><td>宋羽珩</td><td>P</td></walle<>	无	{size(int	Mediu	{size(int):	宋羽珩	P
etCount		t_record>)}	m	8}		
		构造完成						
Wallet_g	getItem()	List <walle< td=""><td>无</td><td>{返回点</td><td>Mediu</td><td>{返回点击</td><td>宋羽珩</td><td>P</td></walle<>	无	{返回点	Mediu	{返回点击	宋羽珩	P
etItem		t_record>		击的列	m	的列表项}		
		构造完成		表项}				
Wallet_g	getItemId(List <walle< td=""><td>无</td><td>{positio</td><td>Mediu</td><td>{position(in</td><td>宋羽珩</td><td>P</td></walle<>	无	{positio	Mediu	{position(in	宋羽珩	P
etItemId)	t_record>		n(int)}	m	t): 5}		
		构造完成						
Wallet_s	setText1()	List <walle< td=""><td>{date:20</td><td>{2017-5-</td><td>Mediu</td><td>{2017-5-21</td><td>宋羽珩</td><td>P</td></walle<>	{date:20	{2017-5-	Mediu	{2017-5-21	宋羽珩	P
etText1		t_record>	17-5-21}	21}(get	m	}		
		构造完成		方法测				
			_	试)				
Wallet_s	setText2()	List <walle< td=""><td>{ "消</td><td>{ "消</td><td>Mediu</td><td>{"消费"}</td><td>宋羽珩</td><td>P</td></walle<>	{ "消	{ "消	Mediu	{"消费"}	宋羽珩	P

etText2		t_record>	费"}	费"}(get	m			
		构造完成		方法测				
				试)				
Wallet_s	setText3()	List <walle< td=""><td>{"消费</td><td>{"消费</td><td>Mediu</td><td>{"消费时</td><td>宋羽珩</td><td>P</td></walle<>	{"消费	{"消费	Mediu	{"消费时	宋羽珩	P
etText3		t_record>	时间"}	时	m	间"}		
		构造完成		间"}(get				
				方法测				
				试)				
Wallet_g	获取列表	List <walle< td=""><td>{环境上</td><td>{返回列</td><td>Mediu</td><td>{返回列表</td><td>宋羽珩</td><td>P</td></walle<>	{环境上	{返回列	Mediu	{返回列表	宋羽珩	P
etView	试图	t_record>	下文:	表试图}	m	试图}		
		构造完成	context}					

3.10 WalletActivity 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Cred_on	列表刷新	onCreate()	{user_id	{amount	Mediu	{amount(Str	宋羽珩	P
Refresh	方法	方法执行	(Stirng):	(String):	m	ing):100.00		
			1337777	100.00}		}		
			6666}					
Cred_cli	返回键监	onCreate()	{点击返	返回主	Mediu	返回主界面	宋羽珩	P
ck	听方法	方法执行	回键}	界面	m			
Cred_on	返回按钮	onCreate()	{点击返	返回主	Mediu	返回主界面	宋羽珩	P
back	监听	方法执行	回按钮}	界面	m			

3.11 ActivityHelpContent 单元测试结果

测试 ID	测试名称	前置条件	输入数	期待输	优	实	执	P/
			据	出	先	际	行	F
					级	输	人	
						出		
Activity	对应的请	无	{int	{向左划	low	{int type:1}	宋羽珩	P
HelpCon	求是否能		type:1}	打开侧				
tent	显示相应			滑菜单,				
	的帮助信			点击"登				
	息			录 / 注				
				册",进				
				入登录				
				界面,在				
				"输入				
				账号"处				
				输入您				
				的已经				
				注册的				
				账号,在				
				" 输 入				
				密码"处				
				输入您				
				的密码,				
				确认无				
				误后点				
				击"登				
				录"按				
				钮,即可				
				成功登				
				录。}				
Activity	返回键监	无	{按下返	{执行返	low	{按下返回	宋羽珩	P
HelpCon	听方法		回键 1}	回主界		键 1}		
tent_bac				面动作}				
k								

3.12 HelpActivity 单元测试结果

测试 ID	测试名称	前	输入	期待输	优先级	实际输	执行人	P/F
		置	数据	出		出		
		条						
		件						
HelpActivity_back	返回键监听方	无	{按下	{执行返	low	{执行返	黎文彬	P
	法		返 回	回主界		回主界		
			键 1}	面动作}		面动作}		

HelpActivity_list	列表点击事件	列表已经加载所有的帮助信息变量	点 点 表 一 。 。	(执行跳 表 项 应 容动作)	Medium	(执行跳列表项应容动作)	黎文彬	P
HelpActivity_list_title	列表标题与内容的对应	列表已经加载所有的帮助信息标题项	(点) 有项 (点) 表一	扶表项应容标列一题动行到第所的内题表项相作跳列一对内容与第标同	Medium	{转表项应容标列一题动行到第所的内题表项相作}跳列一对内容与第标同	黎文彬	P
HelpActivity_list_gettitle	测 试 getTutorialTitle 方法	列表已经加载所有的帮助信息	无	{ List(含 有帮助内容标题)}	high	{ List(含 有帮助 内容标 题)}	黎文彬	P

标			
题			
项			

3.13 ChartActivity 单元测试结果

测试 ID	测试名	前置条	输入数据	期待输	优先级	实际输	执行人	P/F
	称	件		出		出		
ChartActivity_back	返回键	无	{按下返	{执行返	low	{ 执 行	黎文彬	P
	监听方		回键}	回主界		返回主		
	法			面动作}		界面动		
						作}		

3.14 ChartAdapter 单元测试结果

测试 ID	测试名称	前置条	输入数据	期待输出	优先级	实际输出	执 行	P/F
		件					人	
ChartAdapter1	getItem 方法	所有的	{int	{ fragment	High	{ fragment	黎文	P
		fragment	position:	(第二		(第二	彬	
		已经初	1}	个) }		个) }		
		始化完						
		毕						
ChartAdapter2	getCount 方	所有的	{无}	{int : 5}	Low	{int : 5}	黎文	P
	法	fragment					彬	
		已经初						
		始化完						
		毕						
ChartAdapter3	getPageTitle	所有的	{int	{String: 消	High	{String:	黎文	P
	方法	fragment	position:	费金额}		消费金	彬	
		已经初	1}			额}		
		始化完						
		毕						

3.15 ChartFragment 单元测试结果

测试 ID	测试名称	前置条	输入数据	期待输	优先	实际输	执行人	P/F
		件		出	级	出		
ChartFragment1	initControls	无	{View	{界面显	High	{ 界 面	黎文彬	P
	方法		view: 当前	示所有		显示所		

	页 面 的	的列表	2	有的列	
	view}	的标题}		表的标	
				题}	

3.16 ConsumeFragment 单元测试结果

测试 ID	测试名称	前置条	输入数据	期待输出	优先级	实际输出	执 行	P/F
		件					人	
ConsumeFragme	queryData 方	无	{无}	{Vector:	High	{Vector:	黎文	P
nt1	法			data_raw}		data_raw}	彬	
ConsumeFragme	generateDefa	已经从	{无}	将 data_raw	High	{ 将	黎文	P
nt2	ultData 方法	服务器		的数据截		data_raw	彬	
		获得所		取最近7条		的数据截		
		需 数 据		加载在图		取最近 7		
		并装载		表上}		条加载在		
		在				图表上}		
		data_raw						
		中						
ConsumeFragme	resetViewport	无	{无}	{设置了	Low	{设置了	黎文	p
nt3	方法			坐标轴信		坐标轴信	彬	
				息 的 图		息 的 图		
				标}		标}		
ConsumeFragme	onValueSelect	所有的	{ int	{ Snackbar	Medium	{ Snackbar	黎文	p
nt4	ed 方法	数据已	columnInde	提示点击		提示点击	彬	
		经装载	x : 1, int	的 value		的 value		
		在图上	subcolumnI	值}		值}		
			ndex:1,					
			Subcolumn					
			Value					
			value}					

3.17 DistanceUtils 单元测试结果

测试 ID	测试名称	前	输入数据	期待输出	优先	实际输出	执行人	P/F
		置			级			
		条						
		件						
DistanceUtils1	getDistanceOfMeter	己	{double lat1:	{double:	High	{double :	黎文彬	F
	方法	经	30.557498,	1037.9}		1037.9}		
		获	double lng1:					

得	103.998449,
特	
定	double lat2:
目	30.563381,
标	double lng2:
点	104.006849}
的	
经	
纬	
度	

3.18 GetCreditFragment 单元测试结果

测试 ID	测试名称	前置条	输入	期待输出	优先	实际输出	执	行	P/F
		件	数据		级		人		
GetCreditFragment1	queryData	无	{空}	{int[] :	High	{int[] : each_type	黎	文	P
	方法			each_type		(包含从服务器	彬		
				(包含从		请求会的所有数			
				服务器请		据封装得到的数			
				求会的所		组)}			
				有数据封					
				装得到的					
				数组)}					
GetCreditFragment2	Reset 方法	无	{空}	{重置之	Low	{重置之后的图	黎	文	P
				后 的 图		表}	彬		
				表}					
GetCreditFragment3	generateData	已经从服	{空}	{将数据	High	{将数据加载之	黎	文	P
	方法	务器获取		加载之后		后的图表}	彬		
		所需数据		的图表}					
		并且加载							
		在							
		each_type							
		上							

3.19 NetUtils 单元测试结果

测试 ID	测试名称	前置条件	输入数据	期待输出	优先	实际输	执行	P/F
					级	出	人	
NetUtils1	getRequest	获取参数	{final String url:	{执行动	High	{执行动	黎文	P
	方法	param	http://123.206.80.	作:新开		作:新开	彬	

			243:8080/sharing	一个线程		一个线		
			_bicycle/bike_ful	向服务器		程向服		
			1, final	发送查询		务器发		
			Map <string,strin< th=""><th>车辆信息</th><th></th><th>送查询</th><th></th><th></th></string,strin<>	车辆信息		送查询		
			g> params, final	请求}		车辆信		
			Handler handler}			息请求}		
NetUtils2	postRequest	获 取	{final String url:	{执行动	High	{执行动	黎文	P
	方法	String	http://123.206.80.	作: 向服		作: 向服	彬	
		URL和参	243:8080/sharing	务器发送		务器发		
		数 param	_bicycle/finish_p	一个完成		送一个		
			reorder.do, final	预约请求		完成预		
			List <basicname< th=""><th>并获取返</th><th></th><th>约请求</th><th></th><th></th></basicname<>	并获取返		约请求		
			ValuePair>	回的结果		并获取		
			params, final	(含有		返回的		
			Handler handler}	true 或者		结果(含		
				false) }		有 true		
						或 者		
						false) }		

3.20 PreorderHelp 单元测试结果

测试 ID	测试名称	前置条件	输入数据	期待输出	优先 级	实际输出	执行人	P/F
PreorderHelp1	PreorderHel p 方法	无	{无}	{ Preorder Help 对象: time_inter val_help=-1,timeRun nable=null }	High	{ PreorderHe lp 对象: time_interval _help=-1,tim eRunnable=n ull }	黎文彬	P
PreorderHelp2	getPreorder Help 方法	Preorder Help 对象 已经被初 始化	{final String url : http://12 3.206.80 .243:808 0/sharin g_bicycl e/finish_ preorder. do, final	{执信器个约获的有者false)}	High	{执行动作: 向服务完成 预约。 有数数 有数数 有数数 有数数 有数数 有数数 有数数 有数数 有数数 有数	黎文彬	P

			List <bas< th=""><th></th><th></th><th></th><th></th><th></th></bas<>					
			icName					
			ValuePai					
			r>					
			params,					
			final					
			Handler					
			handler}					
PreorderHelp3	getTime_int	Preorder	{空}	{int: 对象	High	{int: 对象的	黎文彬	P
	erval_help	Help 对象		的属性	8	属性	3,7510	
	方法	已经被初		time_inter		time_interval		
		始化		val_help}		_help}		
PreorderHelp4	setTime_inte	无	{int:32}	{无}	High		黎文彬	P
···	rval_help 方	–		· · · · ·		,		
	iven_morp							
PreorderHelp5	getPreorder_	Preorder	{空}	{String: 对	High	{String: 对	黎文彬	P
•	bike_id 方法	Help 对象		象的属性:		象的属性:		
		已经被初		preorder_b		preorder_bik		
		始化		ike_id}		e_id}		
PreorderHelp6	setPreorder_	无	{String	{无}	high	{无}	黎文彬	P
_	bike_id 方法		preorder	-	_			
			_bike_id					
			:"1"}					
PreorderHelp7	getTimeRun	Preorder	{无}	{String: 对	High	{String: 对	黎文彬	P
_	nable 方法	Help 对象		象的属性:		象的属性:		
		已经被初		timeRunna		timeRunnabl		
		始化		ble}		e}		
PreorderHelp8	setTimeRun	无	{Runnab	{无}	High	{无}	黎文彬	P
	nable 方法		le					
			timeRun					
			nable :					
			runnable					
			}					

3.21 PreorderActivity 单元测试结果

测试 ID	测试名称	前置	输入数据	期待输出	优先	实际	执行	P/F
		条件			级	输出	人	
PreorderActivity1	onBackPress	无	{返回键事件}	{执行动作: 跳转	Low	{ 执	黎文	P
	ed 方法			到主界面}		行动	彬	
						作:		

						跳转 到主 界 面}		
PreorderActivity2	onTimeSet ()方法 1	无	{TimePicker view: 当前 view, int hourOfDay: 所选择时间的小时 数 , int minute: 所选择时间的分钟数 }	{将显示的时间 置为所选择的时间}	High	{显的间为选的间 将示时置所择时}	黎文彬	P
PreorderActivity3	onClick () 方 法 (confirmB utton)	已选合的约长经择理预时	无	{执行动作:向服 务器发送请求}	High	{ 行作向务发请求}	黎文彬	P

3.22 MainActivity 单元测试结果

测试 ID	测试名称	前置条件	输入数据	期待输出	优先	实际输出	执行人	P/F
					级			
MainActivity1	onSaveInsta	User 对象	{Bundle	{无}	Low	{无}	黎文彬	F
	nceState 方	不为空,	outState:					
	法	timeremai	系统预设,					
		nning □	Persistable					
		经初始化	Bundle					
			outPersiste					
			ntState: 系					
			统预设}					
MainActivity2	onResume	User 对象	{无}	{无}	High	{无}	黎文彬	P
	方法	不为空,						
		timeremai						
		nning 和						
		mainPreH						
		elp.getTi						
		me_interv						
		al_help都						

		为-1						
MainActivity3	onResume 方法	User 对象 不为空, timeremai nning 或 mainPreH elp.getTi me_interv al_help 不 为-1	{无}	{执行动作:显示倒计时例	High	{执行动作: 显示倒计时 图标并倒计 时}	黎文彬	P
MainActivity4	queryBicycl eLoc 方法	无	{无}	{Vector bicycles}	High	{Vector bicycles}	黎文彬	P
MainActivity5	count_time 方法	预约已经 成功并且 未到达预 约时长	{int interval: 剩余的预约时间秒数}	{执行动 作:显示 倒计时}	High	{执行动作: 显示倒计 时}	黎文彬	P
MainActivity6	findBicycle ByID 方法	无	{String initID: 1}	{Bicycle: 1 号车辆 对象}	Med ium	{Bicycle: 1 号车辆对 象}	黎文彬	P
MainActivity7	Scroll 方法	背景图像 view 已经 初始化	{无}	{执行动作:背景图像规律移动}	Low	{执行动作: 背景图像规 律移动}	黎文彬	P
MainActivity8	onBackPress ed 方法	无	{返回键事件}	{执行动 作:侧滑 菜 单 关 闭}	Low	{执行动作: 侧滑菜单关 闭}	黎文彬	P
MainActivity9	registerWige t 方法	无	{无}	{初始化 控件}	High	{初始化控件}	黎文彬	P
MainActivity10	queryUserW alletAndCre dit 方法	无	{String ID : 123456789 11}	{double[] :账号为 123456 78911 对应的 余额和 积分数 组}	High	{double[]: 账号为 12345678 911 对应 的余额和 积分数 组}	黎文彬	P
MainActivity11	proccessAcc ount 方法	用户已经 登录	{String account: 123456789 11}	{String: 123*** *8911}	Low	{String : 123****8 911}	黎文彬	P
MainActivity12	addMarkerT	已经获得	{空}	{将车辆	High	{将车辆表	黎文彬	P

oMap 方法	车辆位置	表示在	示在	地图	
	信息	地图	上}		
		上}			

4 对软件功能的结论

4.1 功能

为了开发出真正满足用户需求的软件产品,首先必须知道用户的需求,必须使用适当方法与用户沟通,根据共享单车目前的实际使用情况,系统应当包括登录模块、注册模块、预约用车模块、查看行程模块、远程开锁模块、故障申报模块、显示位置信息模块、数据分析模块、导航模块、使用帮助模块、查看钱包模块、积分功能模块十二大功能模块,具体描述为以下:

- 1. 登录模块功能: 用户帐号、密码的验证,提供向注册模块的入口。
- 2. 注册模块功能:用户账号是否符合规范、密码输入及确认、账号和密码记录入数据库、查询账号是否可用、提供向登录模块的入口。
- 3. 预约用车模块:获取用户当前位置、选择车辆、选择合适的预约开始和结束时间,进行 预约并同步至数据库
- 4. 查看行程模块: 获取用户所有的行程统计
- 5. 远程开锁模块功能:获取用户当前位置,选择车辆,验证用户与车辆的距离信息,将数据同步至数据库,计时功能。
- 6. 故障申报模块功能:选择车辆,获取车辆 ID,获取用户输入的报修信息,并同步至数据库,为用户增加积分
- 7. 显示位置信息模块功能: 获取用户位置信心, 并显示用户的位置并显示在画面中央。
- 8. 数据分析模块: 获取用户的车辆统计信息并以饼状图显示、获取用户最近 7 天的消费金额并以柱状图显示、获取用户最近 7 天的积分变化信息并以折线图显示、获取用户最近 7 天的积分获得信息并以饼状图显示、获取用户的周统计信息并以柱状图显示。
- 9. 导航模块功能: 获取用户实时位置,并获取用户输入的目的地,提供规划路线,模拟导航。
- 10. 使用帮助模块功能:显示相应的教程和疑问解答。
- 11. 查看钱包模块功能: 获取用户的消费记录并以列表显示。
- 12. 查看积分模块功能: 获取用户的积分拜年话记录并以列表显示。
- 13. 系统运行在 Android 平台上,要求有一个较好的图形用户界面,操作要求简单。

4.2 界面

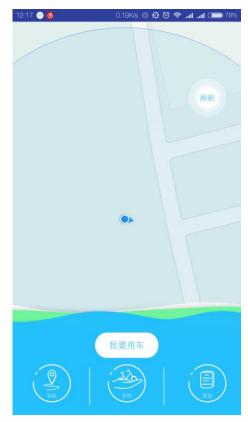


图 1. 用车界面



图 2. 侧滑界面



图 3. 登录界面



- 2017-05-24 22:18
 - 消费: -0.02
 - 消费时间: 2017-05-24 22:18
- 2017-05-24 21:09
 - 消费: -0.03
 - 消费时间: 2017-05-24 21:09
- ◆ 2017-05-24 21:09 消费: -0.03
 - 消费时间: 2017-05-24 21:09
- 2017-05-24 20:35
 - 消费: -0.02 ■ 消费时间: 2017-05-24 20:35
- ◆ 2017-05-24 20:09 消费: -0.03
 - 消费时间: 2017-05-24 20:09
- 2017-05-24 20:06
- 消费: -0.02
- 消费时间: 2017-05-24 20:06
- ◆ 2017-05-24 20:06 消费: -0.02
 - 消费时间: 2017-05-24 20:06
- 2017-05-24 19:50
 - 消费: -0.03
 - 消费时间: 2017-05-24 19:50

图 4. 账单详情界面



图 5. 注册界面图



- 2017-05-24 22:18
 - 积分变化: 5.0
- 备注:骑行消费获得的积分
- 2017-05-24 21:09
 - 积分变化: 5.0 ■ 备注: 骑行消费获得的积分
- 2017-05-24 21:09
 - 积分变化: 5.0
- 备注: 骑行消费获得的积分
- ◆ 2017-05-24 20:35 积分变化: 5.0
 - 备注: 骑行消费获得的积分
- ◆ 2017-05-24 20:09 积分变化: 5.0
 - 备注: 骑行消费获得的积分
- 2017-05-24 20:06
- 积分变化: 5.0
- 备注: 骑行消费获得的积分
- ◆ 2017-05-24 20:06 积分变化: 5.0
 - 备注: 骑行消费获得的积分
- ◆ 2017-05-24 19:50 积分变化: 5.0
 - 备注: 骑行消费获得的积分
- 2017-05-24 19:43

图 6. 积分详情界面

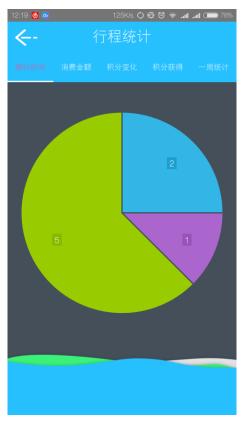


图 7. 行程统计界面

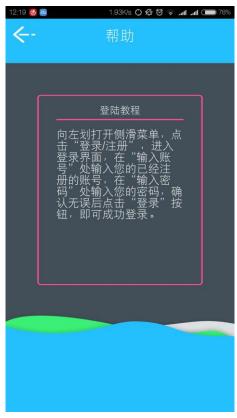


图 8. 帮助界面

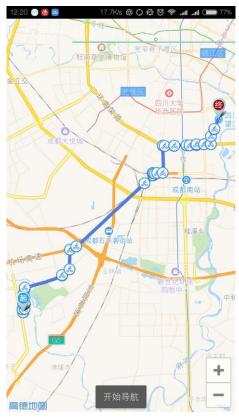


图 9. 导航界面



图 10. 预约界面



图 11. 报修界面截图

4.3 能力

功能名称	软件能力	正确性	可操作性	备注
用户登录	用户使用正 确的账号和 密码登录	正确	可操作	不可扩展
用户注册	用户使用为 注册的账户 和密码进行 注册	正确	可操作	不可扩展
预约用车	对未被使用 和预约的车 辆进行预约	正确	可操作	不可扩展
查看行程	获取用户行 程数据	正确	可操作	不可扩展
远程开锁	获取用户定 位和计时	正确	可操作	不可扩展
故障申报	获取车辆 ID 记录用 户报修描述	正确	可操作	可扩展
显示位置信	获取用户和	正确	可操作	可扩展

息	车辆位置信 息并显示			
数据分析	获取用户骑 行和积分、 消费信息	正确	可操作	可扩展
导航模块	获取用户定 位和路线规 划	正确	可操作	可扩展
使用帮助模块	提供教程和 常见问题解 答	正确	可操作	可扩展
查看钱包模 块	获取用户消 费记录	正确	可操作	不可扩展
查看积分	获取用户积 分记录	正确	可操作	不可扩展

5分析摘要

5.1 缺陷和限制

本软件基本达到预期的功能需求,能够实现用车,还车,远程开锁等功能。但由于开发时间有限,对软件的用户体验考虑较少,比如:未加入免登陆功能。在每次重新启用软件时,需要重新登录。软件对性能考虑还有待进一步改善,在单元测试时,由于是对软件核心代码进行逐一测试,未发现性能上的问题。在实际运行时,软件会有短暂卡顿现象。软件的安全方面,由于软件设计到用户隐私数据,对数据的安全性要求高,在这一方面还有改进空间。

本软件是基于 C/S (客户端/服务器) 架构进行开发的, 所以对网络要求高, 软件在设计环节, 未考虑网络状况对软件的影响, 有时会由于网络状况, 引发界面停止跳转。

5.2 评价

本软件实现了需求清单中所有功能,符合软件预期标准。软件由于资源图片过大等原因,有时会导致资源空间不足,以及内存占用过大的现象出现,会导致软件卡顿的现象出现。在软件的性能上还有待进一步优化。

6 测试资源消耗

- 1. 系统运行环境: Android 5.0以上操作系统(客户端) Unbuntu Server (服务器)
- 2. 系统开发平台: Android Studio MyEclipse
- 3. 数据库管理系统软件: MySQL 5.7
- 4. 系统测试用时:三天
- 5. 系统测试人员: 宋羽珩 黎文彬