

《sharing-bicycle》单元测试报告

宋羽珩 黎文彬

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	9
2.13 ChartActivity 单元测试设计	9
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	10
2.17 DistanceUtils 单元测试设计	11
2.18 GetCreditFragment 单元测试设计	11
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	15
3.2 Credit_record 单元测试结果	16
3.3 CreditActivity 单元测试结果	17
3.4 CreditAdapter 单元测试结果	18
3.5 LoginActivity 单元测试结果	18
3.6 NavigateActivity 单元测试结果	19
3.7 NetUtils 单元测试结果	19
3.8 Wallet_record 单元测试结果	21
3.9 WalletAdapter 单元测试结果	21
3.10 WalletActivity 单元测试结果	22
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	测试名称	前置条件	输入数据	期待输	优先级
-------	------	------	------	-----	-----

				出	
Bic-getBike_id	Id_get 方法	构造 Bicycle 对象	{bike_id:1}	{1}	Low
Bic_setBike_id	Id_set 方法	构造 Bicycle 对象	{bike_id:5}	{5}(get 方法测试)	Low
Bic_getLocation	Location_get 方法	构造 Bicycle 对象	{location:(104.001904, 30.560536)}	{(104.001904,30.560536)}	Low
Bic_setLocation	Location_set 方法	构造 Bicycle 对象	{location:(103.99655,30.565947)}	{(103.99655,30.565947) } (get 方法测试)	Low
Bic_isInuse	Inuse_get 方法	构造 Bicycle 对象	{in_use(boolean):false}	{false}	Low
Bic_setInuse	Inuse_set 方法	构造 Bicycle 对象	{in_use(boolean):true}	{true}(get 方法测试)	Low
Bic_isBreakdown	Isbreakdown_get 方法	构造 Bicycle 对象	{isbreakdown(boolean):true}	{true}	Low
Bic_setBreakdown	Breakdown_set 方法	构造 Bicycle 对象	{ isbreakdown(boolean):false }	{false} (get 方法测试)	Low
Bic_Id_preorder	Idpreorder_get 方法	构造 Bicycle 对象	{id_preorder(boolean):false}	{false}	Low
Bic_setId_preorder	Idpreorder_set 方法	构造 Bicycle 对象	{id_preorder(boolean):true}	{true}(get 方法测试)	Low
Bic_getDescription	Description_get	构造 Bicycle 对象	{description(String): 车辆故障}	{车辆故障}	Low
Bic_setDescription	Description_set	构造 Bicycle 对象	{description(String): 车胎没气}	{车胎没气}	Low

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_getamount	getCredit_amount()	构造 credit_record 对象	{credit_amount(String): 100}	{100}	Low

Cre_setamount	setCredit_amount()	构造 credit_record 对象	{credit_amount(String): 95}	{95}	Low
Cre_getdate	getDate_time()	构造 credit_record 对象	{date_time(String):2017-5-20}	{2017-5-20}	Low
Cre_setdate	setDate_time()	构造 credit_record 对象	{date_time(String):2017-5-26}	{2017-5-26}	Low
Cre_getdescription	getDescription()	构造 credit_record 对象	{description(String): 骑行奖励}	{骑行奖励}	Low
Cre_setDescription	setDescription()	构造 credit_record 对象	{description(String): 报修奖励}	{报修奖励}	Low

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_onRefresh	列表刷新方法	onCreate() 方法执行	{下拉列表}	返回列表数据	Medium
Cred_click	返回键监听方法	onCreate() 方法执行	{点击返回键}	返回主界面	Medium
Cred_onback	返回按钮监听	onCreate() 方法执行	{点击返回按钮}	返回主界面	Medium

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_getCount	getCount()	List<credit_record>构造完成	无	{size(int)}	Medium
Credit_getItem	getItem()	List<credit_record>构造完成	无	{返回点击的列表项}	Medium
Credit_getItemId	getItemId()	List<credit_record>构造完成	无	{position(int)}	Medium
Credit_setText1	setText1()	List<credit_record>构造完成	{date:2017-5-21}	{2017-5-21}(get 方法测试)	Medium
Credit_setText2	setText2()	List<credit_record>构造完成	{“积分变化”}	{“积分变	Medium

Text2		d>构造完成		化”}(get 方法测试)	um
Credit_set Text3	setText3()	List<credit_recor d>构造完成	{ “备注” }	{ “备注” }(get 方法测试)	Medi um
Credit_get View	获取列表试 图	List<credit_recor d>构造完成	{环境上下文: context}	{ 返回列表 试图}	Medi 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 ry	查询用户积 分方法	用户登录后	{user_id(Stirng):13577 776666}	{balance(do uble),credit(double)}	Medi um
Login_clic k	登录按钮方 法	无	{id(String) : 13577776666,password (String): root123}	{flag:true}	High
Login_reg ister	注册界面跳 转	无	点击注册按钮	跳转到注册 界面	Medi um
Login_bac k	返回键监听	无	点击返回键	返回主界面	Medi 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 etInput	搜索目的地 方法	地图视图	{query(String):成都}	显示搜索列 表	Medi um
Navi_onIt mclick	列表项点击 方法	地图视图	{Itemname(String): 四 川大学(望江校区) }	显示当前位 置到目的地 的路径	Medi um
Navi_back	返回监听方 法	无	无	跳 转 Mainactivit y	Medi um

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 方法	无	{url(String): http://123.206.80.243:8 080/sharing_bicycle/fi nish_preorder.do;parm suserid(String):135777 76666 , bike_id(String):1}	{success(Str ing)}	High

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 mount	getAmount()	构 造 wallet_record 对 象	{amount(String):100.0 0}	100.00	Low
Wall_seta mount	setAmount()	构 造 wallet_record 对 象	{amount(String):99.5}	99.5	Low
Wall_getd ate	getDate_tim e()	构 造 wallet_record 对 象	{date_time(String):201 7-5-25}	{2017-5-25 }	Low
Wall_setd ate	setDate_tim e()	构 造 wallet_record 对 象	{date_time(String):201 7-5-27}	{2017-5-27 }	Low

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 Count	getCount()	List<wallet_reco rd>构造完成	无	{size(int)}	Medi um
Wallet_get Item	getItem()	List<wallet_reco rd>构造完成	无	{ 返回 点击 的列表项}	Medi um
Wallet_get ItemId	getItemId()	List<wallet_reco rd>构造完成	无	{position(in t)}	Medi um
Wallet_set Text1	setText1()	List<wallet_reco rd>构造完成	{date:2017-5-21}	{2017-5-21 }(get 方法 测试)	Medi um

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

2.10 WalletActivity 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
Cred_onR efresh	列表刷新方 法	onCreate() 方 法 执行	{user_id(Strrng):13377 776666}	{amount(Str ing):100.00 }	Medi um
Cred_clic k	返回键监听 方法	onCreate() 方 法 执行	{点击返回键}	返回主界面	Medi um
Cred_onb ack	返回按钮监 听	onCreate() 方 法 执行	{点击返回按钮}	返回主界面	Medi 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 菜 单,点击“登 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。}	low

ActivityHelpContent_back	返回键监听方法	无	{按下返回键 1}	{执行返回主界面动作}	low
--------------------------	---------	---	-----------	-------------	-----

2.12 HelpActivity 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
HelpActivity_back	返回键监听方法	无	{按下返回键 1}	{执行返回主界面动作}	low
HelpActivity_list	列表点击事件	列表已经加载所有的帮助信息变量	{点击列表第一项}	{执行跳转到列表第一项所对应的内容动作}	Medium
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	{点击列表第一项}	{执行跳转到列表第一项所对应的内容，内容标题与列表第一项标题相同动作}	Medium
HelpActivity_list_gettitle	测试 getTutorialTitle 方法	列表已经加载所有的帮助信息标题项	无	{List(含有帮助内容标题)}	high

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	返回键监听方法	无	{按下返回键}	{执行返回主界面动作}	low

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 fragment 已经初始化完毕	{int position: 1}	{ fragment (第 二个) }	High
ChartAdapter2	getCount 方法	所有的 fragment 已经初始化完毕	{无}	{int : 5}	Low
ChartAdapter3	getPageTitle 方法	所有的 fragment 已经初始化完毕	{int position: 1}	{String : 消费金额}	High

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 方法	无	{View view: 当前页面的 view}	{界面显示所有的列表的标题}	High

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 : data_raw}	High
ConsumeFragment2	generateDefaultData 方法	已经从服务器获得所需数据并装载在 data_raw 中	{无}	{ 将 data_raw 的数据截取最近 7 条加载在图表上}	High
ConsumeFragment3	resetViewport 方法	无	{无}	{ 设置了坐标轴信息的图标}	Low
ConsumeFragment4	onValueSelected 方法	所有的数据已经装载在图上	{ int columnIndex : 1, int subcolumnIndex:1,	{ Snackbar 提示点击的 value 值}	Medium

			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 方法	已经获得特定目标点的经纬度	{double lat1 : 30.557498, double lng1 : 103.998449, double lat2 : 30.563381, double lng2 : 104.006849}	{double: 1037.9}	High

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[] : each_type (包含从服务器请求会的所有数据封装得到的数组)}	High
GetCreditFragment2	Reset 方法	无	{空}	{重置之后的图表}	Low
GetCreditFragment3	generateData 方法	已经从服务器获取所需数据并且加载在 each_type 上	{空}	{将数据加载之后的图表}	High

2.19 NetUtils 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
NetUtils1	getRequest 方法	获取参数 param	{final String url: http://123.206.80.243:808	{执行动作: 新开	High

			0/sharing_bicycle/bike_full, final Map<String,String> params, final Handler handler}	一个线程 向服务器 发送查询 车辆信息 请求}	
NetUtils2	postRequest 方法	获 取 String URL和参 数 param	{final String url : http://123.206.80.243:8080/sharing_bicycle/finish_preorder.do, final List<BasicNameValuePair> params, final Handler handler}	{ 执行动作：向服务器发送一个完成预约请求并获取返回的结果（含有true 或者false）}	High

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	PreorderHelp 方法	无	{无}	{ PreorderHelp 对象 : time_interval_help =-1,time Runnable =null }	High
PreorderHelp2	getPreorderHelp 方法	PreorderHelp 对象已经被初始化	{final String url : http://123.206.80.243:8080/sharing_bicycle/finish_preorder.do, final List<BasicNameValuePair> params, final Handler handler}	{ 执行动作：向服务器发送一个完成预约请求并获取返回的结果（含有true 或者false）}	High
PreorderHelp3	getTime_interval_help	PreorderHelp 对象	{空}	{int: 对象的属性}	High

	方法	已经被初始化		time_interval_help }	
PreorderHelp4	setTime_interval_help 方法	无	{int:32}	{无}	High
PreorderHelp5	getPreorder_bike_id 方法	Preorder Help 对象已经被初始化	{空}	{String : 对象的属性 : preorder_bike_id}	High
PreorderHelp6	setPreorder_bike_id 方法	无	{String preorder_bike_id:"1"}	{无}	high
PreorderHelp7	getTimeRunnable 方法	Preorder Help 对象已经被初始化	{无}	{String : 对象的属性 : timeRunnable}	High
PreorderHelp8	setTimeRunnable 方法	无	{Runnable timeRunnable: runnable}	{无}	High

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	onBackPressed 方法	无	{返回键事件}	{ 执行动作：跳转到主界面}	Low
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
PreorderActivity3	onClick () 方法 (confirmButton)	已经选择合理的预约时长	无	{ 执行动作：向服务器发送请求}	High

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	onSaveInstanceState 方法	User 对象不为空, timeremaining 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, timeremaining 和 mainPreHelp.getTimeInterval_help 都为-1	{无}	{无}	High
MainActivity3	onResume 方法	User 对象不为空, timeremaining 或 mainPreHelp.getTimeInterval_help 不为-1	{无}	{执行动作: 显示倒计时图标并倒计时}	High
MainActivity4	queryBicycleLoc 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	findBicycleByID 方法	无	{String initID: 1}	{Bicycle: 1 号车辆对象}	Medium
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	onBackPressed 方法	无	{返回键事件}	{执行动作: 侧滑菜单关闭}	Low
MainActivity9	registerWidget 方法	无	{无}	{初始化控件}	High
MainActivity10	queryUserW	无	{String ID :	{double[]}	High

	alletAndCredit 方法		12345678911}	: 账号为 12345678911 对应的余额和积分数组}	
MainActivity11	processAccount 方法	用户已经登录	{String account : 12345678911}	{String : 123***8911}	Low
MainActivity12	addMarkerToMap 方法	已经获得车辆位置信息	{空}	{将车辆表示在地图上}	High

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	输入数据	期待输出	优先级	实际输出	执行人	P/F
Bic-getBike_id	Id_get 方法	构造 Bicycle 对象	{bike_id :1}	{1}	Low	{1}	宋羽珩	P
Bic_setBike_id	Id_set 方法	构造 Bicycle 对象	{bike_id :5}	{5}(get 方法测试)	Low	{5}	宋羽珩	P
Bic_getLocation	Location_get 方法	构造 Bicycle 对象	{location:(104.001904,30.560536)}	{{ 104.001904,30.560536}}	Low	{{ 104.001904,30.560536}}	宋羽珩	P
Bic_setLocation	Location_set 方法	构造 Bicycle 对象	{location:(103.9655,30.565947)}	{{ 103.9655,30.565947}}	Low	{{ 103.9655,30.565947}}	宋羽珩	P

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

3.2 Credit_record 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
Cre_geta mount	getCredit_ amount()	构造 credit_rec ord 对象	{credit_ amount(String):	{100}	Low	{100}	宋羽珩	P

			100}					
Cre_setamount	setCredit_amount()	构造 credit_record 对象	{credit_amount(String): 95}	{95}	Low	{95}	宋羽珩	P
Cre_getdate	getDate_time()	构造 credit_record 对象	{date_time(String):2017-5-20}	{2017-5-20}	Low	{2017-5-20}	宋羽珩	P
Cre_setdate	setDate_time()	构造 credit_record 对象	{date_time(String):2017-5-26}	{2017-5-26}	Low	{2017-5-26}	宋羽珩	P
Cre_getdescription	getDescription()	构造 credit_record 对象	{description(String): 骑行奖励}	{骑行奖励}	Low	{骑行奖励}	宋羽珩	P
Cre_setDescription	setDescription()	构造 credit_record 对象	{description(String): 报修奖励}	{报修奖励}	Low	{报修奖励}	宋羽珩	P

3.3 CreditActivity 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
Cred_onRefresh	列表刷新方法	onCreate() 方法执行	{下拉列表}	返回列表数据	Medium	列表数据	宋羽珩	P
Cred_click	返回键监听方法	onCreate() 方法执行	{点击返回键}	返回主界面	Medium	成功跳转	宋羽珩	P
Cred_onback	返回按钮监听	onCreate() 方法执行	{点击返回按钮}	返回主界面	Medium	成功跳转	宋羽珩	P

3.4 CreditAdapter 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
Credit_getCount	getCount()	List<credit_record>构造完成	无	{size(int)}	Medium	{size(int):6}	宋羽珩	P
Credit_getItem	getItem()	List<credit_record>构造完成	无	{返回点击的列表项}	Medium	{返回点击的列表项}	宋羽珩	P
Credit_getItemId	getItemId()	List<credit_record>构造完成	无	{position(int)}	Medium	{position(int):5}	宋羽珩	P
Credit_setText1	setText1()	List<credit_record>构造完成	{date:2017-5-21}	{2017-5-21}(get方法测试)	Medium	{2017-5-21}	宋羽珩	P
Credit_setText2	setText2()	List<credit_record>构造完成	{“积分变化”}	{“积分变化”}(get方法测试)	Medium	{“积分变化”}	宋羽珩	P
Credit_setText3	setText3()	List<credit_record>构造完成	{“备注”}	{“备注”}(get方法测试)	Medium	{“备注”}	宋羽珩	P
Credit_getView	获取列表视图	List<credit_record>构造完成	{环境上下文: context}	{返回列表视图}	Medium	{返回列表视图}	宋羽珩	P

3.5 LoginActivity 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
Login_query	查询用户积分方法	用户登录后	{user_id(String):1357776666}	{balance(double), credit(double)}	Medium	{balance(double):100.00, credit(double):}	宋羽珩	P

			}			100}		
Login_click	登录按钮方法	无	{id(String) : 13577776666,password (String) : root123 }	{flag:true }	High	{flag:true }	宋羽珩	P
Login_register	注册界面跳转	无	点击注册按钮	跳转到注册界面	Medium	成功跳转	宋羽珩	P
Login_back	返回键监听	无	点击返回键	返回主界面	Medium	成功跳转	宋羽珩	P

3.6 NavigateActivity 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
Navi_onGetInput	搜索目的地方法	地图视图	{query(String): 成都}	显示搜索列表	Medium	{result[]: 列表 (含 北京) }	宋羽珩	P
Navi_onItemclick	列表项点击方法	地图视图	{Itemname(String): 四川大学 (望江校区) }	显示当前位置到目的地的路径	Medium	显示当前位置到四川大学望江校区的路径	宋羽珩	P
Navi_back	返回监听方法	无	无	跳转到Mainactivity	Medium	成功跳转	宋羽珩	P

3.7 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	优先级	实际输出	执行人	P/F
-------	------	------	------	------	-----	------	-----	-----

Net_get	GET 请求方法	无	{url(String): "http://123.206.80.243:8080/sharing_bicycle/wallet_credit.do;parmsuserid(String):1357776666}	{credit(String):100}	High	{credit(String):100}	宋羽珩	P
Net_post	Post 请求方法	无	{url(String): http://123.206.80.243:8080/sharing_bicycle/finish_preorder.do;parmsuserid(String):1357776666 , bike_id(String):1}	{success(String)}	High	有缺陷，参数不完整，无返回数据	宋羽珩	F

3.8 Wallet_record 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
Wall_get amount	getAmount()	构造 wallet_record 对象	{amount(String):100.00}	100.00	Low	100.00	宋羽珩	P
Wall_set amount	setAmount()	构造 wallet_record 对象	{amount(String):99.5}	99.5	Low	99.5	宋羽珩	P
Wall_get date	getDate_time()	构造 wallet_record 对象	{date_time(String):2017-5-25}	{2017-5-25}	Low	{2017-5-25}	宋羽珩	P
Wall_set date	setDate_time()	构造 wallet_record 对象	{date_time(String):2017-5-27}	{2017-5-27}	Low	{2017-5-27}	宋羽珩	P

3.9 WalletAdapter 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
Wallet_getCount	getCount()	List<wallet_record> 构造完成	无	{size(int)}	Medium	{size(int): 8}	宋羽珩	P
Wallet_getItem	getItem()	List<wallet_record> 构造完成	无	{返回点击的列表项}	Medium	{返回点击的列表项}	宋羽珩	P
Wallet_getItemId	getItemId()	List<wallet_record> 构造完成	无	{position(int)}	Medium	{position(int): 5}	宋羽珩	P
Wallet_setText1	setText1()	List<wallet_record> 构造完成	{date:2017-5-21}	{2017-5-21}(get 方法测试)	Medium	{2017-5-21}	宋羽珩	P
Wallet_setText2	setText2()	List<wallet_record> 构造完成	{“消费”}	{“消费”}	Medium	{“消费”}	宋羽珩	P

etText2		t_record> 构造完成	费” } }	费”}(get 方法测试)	m			
Wallet_setText3	setText3()	List<wallet_record> 构造完成	{ “消费时间” }	{ “消费时间”}(get 方法测试)	Medium	{ “消费时间” }	宋羽珩	P
Wallet_getView	获取列表 试图	List<wallet_record> 构造完成	{ 环境上下文 : context }	{ 返回列表试图 }	Medium	{ 返回列表试图 }	宋羽珩	P

3.10 WalletActivity 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
Cred_onRefresh	列表刷新方法	onCreate() 方法执行	{user_id (String): 13377776666}	{amount (String): 100.00}	Medium	{amount(String):100.00 }	宋羽珩	P
Cred_click	返回键监听方法	onCreate() 方法执行	{ 点击返回键 }	返回主界面	Medium	返回主界面	宋羽珩	P
Cred_onback	返回按钮监听	onCreate() 方法执行	{ 点击返回按钮 }	返回主界面	Medium	返回主界面	宋羽珩	P

3.11 ActivityHelpContent 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
Activity HelpContent	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 菜 单, 点 击 “ 登 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。}	low	{int type:1}	宋羽珩	P
Activity HelpContent_back	返回键监听方法	无	{按下返回键 1}	{执行返回主界面动作}	low	{按下返回键 1}	宋羽珩	P

3.12 HelpActivity 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
HelpActivity_back	返回键监听方法	无	{按下返回键 1}	{执行返回主界面动作}	low	{执行返回主界面动作}	黎文彬	P

HelpActivity_list	列表点击事件	列表已经加载所有的帮助信息变量	{ 点击列表第一项}	{执行跳转到列表第一项所对应的内容动作}	Medium	{执行跳转到列表第一项所对应的内容动作}	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	{ 点击列表第一项}	{执行跳转到列表第一项所对应的内容, 内容标题与列表第一项标题相同动作}	Medium	{执行跳转到列表第一项所对应的内容, 内容标题与列表第一项标题相同动作}	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
ChartActivity_back	返回键监听方法	无	{ 按下返回键 }	{ 执行返回主界面动作 }	low	{ 执行返回主界面动作 }	黎文彬	P

3.14 ChartAdapter 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
ChartAdapter1	getItem 方法	所有的 fragment 已经初始化完毕	{int position: 1}	{ fragment (第二个) }	High	{ fragment (第二个) }	黎文彬	P
ChartAdapter2	getCount 方法	所有的 fragment 已经初始化完毕	{无}	{int : 5}	Low	{int : 5}	黎文彬	P
ChartAdapter3	getPageTitle 方法	所有的 fragment 已经初始化完毕	{int position: 1}	{String: 消费金额}	High	{String : 消费金额}	黎文彬	P

3.15 ChartFragment 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
ChartFragment1	initControls 方法	无	{View view: 当前	{界面显示所有	High	{界面显示所	黎文彬	P

			页 面 的 view}	的 列 表 的标题}		有的列 表的标 题}		
--	--	--	----------------	---------------	--	------------------	--	--

3.16 ConsumeFragment 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
ConsumeFragme nt1	queryData 方 法	无	{无}	{Vector : data_raw}	High	{Vector : data_raw}	黎 文 彬	P
ConsumeFragme nt2	generateDefa ultData 方法	已 经 从 服 务 器 获 得 所 需 数 据 并 装 载 在 data_raw 中	{无}	将 data_raw 的 数 据 截 取最近 7 条 加 载 在 图 表 上}	High	{ 将 data_raw 的 数 据 截 取最近 7 条 加 载 在 图 表 上}	黎 文 彬	P
ConsumeFragme nt3	resetViewport 方法	无	{无}	{ 设 置 了 坐 标 轴 信 息 的 图 标}	Low	{ 设 置 了 坐 标 轴 信 息 的 图 标}	黎 文 彬	p
ConsumeFragme nt4	onValueSelect ed 方法	所 有 的 数 据 已 经 装 载 在 图 上	{ int columnInde x : 1, int subcolumnI ndex:1, Subcolumn Value value}	{ Snackbar 提示 点 击 的 value 值}	Medium	{ Snackbar 提示 点 击 的 value 值}	黎 文 彬	p

3.17 DistanceUtils 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
DistanceUtils1	getDistanceOfMeter 方法	已 经 获	{double lat1: 30.557498, double lng1:	{double: 1037.9}	High	{double : 1037.9}	黎文彬	F

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

3.18 GetCreditFragment 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
GetCreditFragment1	queryData 方法	无	{空}	{int[] : each_type (包含从服务器请求会的所有数据封装得到的数组)}	High	{int[] : each_type (包含从服务器请求会的所有数据封装得到的数组)}	黎 文 彬	P
GetCreditFragment2	Reset 方法	无	{空}	{重置之后的图表}	Low	{重置之后的图表}	黎 文 彬	P
GetCreditFragment3	generateData 方法	已经从服务器获取所需数据并且加载在 each_type 上	{空}	{将数据加载之后的图表}	High	{将数据加载之后的图表}	黎 文 彬	P

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	优先级	实际输出	执行人	P/F
NetUtils1	getRequest 方法	获取参数 param	{final String url: http://123.206.80.	{执行动作: 新开	High	{执行动作: 新开	黎文彬	P

			243:8080/sharing_bicycle/bike_full, final Map<String,String> params, final Handler handler}	一个线程向服务器发送查询车辆信息请求}		一个线程向服务器发送查询车辆信息请求}		
NetUtils2	postRequest 方法	获取 String URL 和参数 param	{final String url: http://123.206.80.243:8080/sharing_bicycle/finish_preorder.do, final List<BasicNameValuePair> params, final Handler handler}	{执行动作: 向服务器发送一个完成预约请求并获取返回的结果 (含有 true 或者 false) }	High	{执行动作: 向服务器发送一个完成预约请求并获取返回的结果 (含有 true 或者 false) }	黎文彬	P

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	PreorderHelp 方法	无	{无}	{ PreorderHelp 对象 : time_interval_help=-1,timeRunnable=null }	High	{ PreorderHelp 对象 : time_interval_help=-1,timeRunnable=null }	黎文彬	P
PreorderHelp2	getPreorderHelp 方法	PreorderHelp 对象已经被初始化	{final String url : http://123.206.80.243:8080/sharing_bicycle/finish_preorder.do, final	{执行动作: 向服务器发送一个完成预约请求并获取返回的结果 (含有 true 或者 false) }	High	{执行动作: 向服务器发送一个完成预约请求并获取返回的结果 (含有 true 或者 false) }	黎文彬	P

			List<BasicNameValuePair> params, final Handler handler}					
PreorderHelp3	getTime_interval_help 方法	Preorder Help 对象已经被初始化	{空}	{int: 对象的属性 time_interval_help}	High	{int: 对象的属性 time_interval_help}	黎文彬	P
PreorderHelp4	setTime_interval_help 方法	无	{int:32}	{无}	High	{无}	黎文彬	P
PreorderHelp5	getPreorder_bike_id 方法	Preorder Help 对象已经被初始化	{空}	{String: 对象的属性: preorder_bike_id}	High	{String : 对象的属性: preorder_bike_id}	黎文彬	P
PreorderHelp6	setPreorder_bike_id 方法	无	{String preorder_bike_id :”1”}	{无}	high	{无}	黎文彬	P
PreorderHelp7	getTimeRunnable 方法	Preorder Help 对象已经被初始化	{无}	{String: 对象的属性: timeRunnable}	High	{String : 对象的属性: timeRunnable}	黎文彬	P
PreorderHelp8	setTimeRunnable 方法	无	{Runnable timeRunnable : runnable }	{无}	High	{无}	黎文彬	P

3.21 PreorderActivity 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
PreorderActivity1	onBackPressed 方法	无	{返回键事件}	{执行动作: 跳转到主界面}	Low	{ 执行动作:	黎文彬	P

						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	{ 将显示的时间置为所选择的时间 }	黎文彬	P
PreorderActivity3	onClick () 方 法 (confirmButton)	已经选择合理的预约时长	无	{执行动作: 向服务器发送请求}	High	{ 执行动作: 向服务器发送请求 }	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	优先级	实际输出	执行人	P/F
MainActivity1	onSaveInstanceState 方法	User 对象不为空, timeremaining 已经初始化	{Bundle outState: 系统预设, Persistable Bundle outPersistantState: 系统预设}	{无}	Low	{无}	黎文彬	F
MainActivity2	onResume 方法	User 对象不为空, timeremaining 和 mainPreHelp.getTimeInterval_help 都	{无}	{无}	High	{无}	黎文彬	P

		为-1						
MainActivity3	onResume 方法	User 对象不为空, timerremaining 或 mainPreHelp.getTimeIntervalHelp 不为-1	{无}	{ 执 行 动 作: 显 示 倒 计 时 图 标 并 倒 计 时 }	High	{执行动作: 显示倒计时图标并倒计时}	黎文彬	P
MainActivity4	queryBicycleLoc 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	findBicycleByID 方法	无	{String initID: 1}	{Bicycle: 1 号车辆对象}	Medium	{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	onBackPressed 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	registerWidget 方法	无	{无}	{ 初 始 化 控 件 }	High	{初始化控件}	黎文彬	P
MainActivity10	queryUserWalletAndCredit 方法	无	{String ID : 12345678911}	{double[] : 账号为 12345678911 对应的余额和积分数组}	High	{double[]: 账号为 12345678911 对应的余额和积分数组}	黎文彬	P
MainActivity11	processAccount 方法	用户已经登录	{String account : 12345678911}	{String : 123***8911}	Low	{String : 123****8911}	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 信息		表示在 地 图 上}		示在地图 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

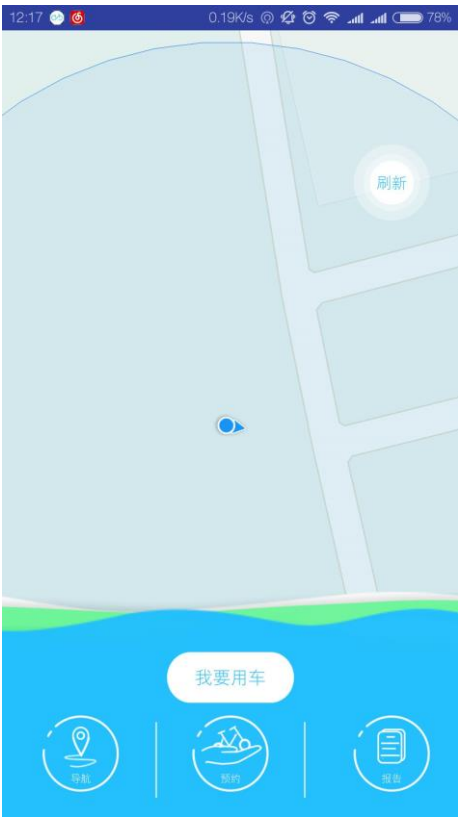


图 1. 用车界面



图 2. 侧滑界面



图 3. 登录界面



图 4. 账单详情界面



图 5. 注册界面图



图 6. 积分详情界面

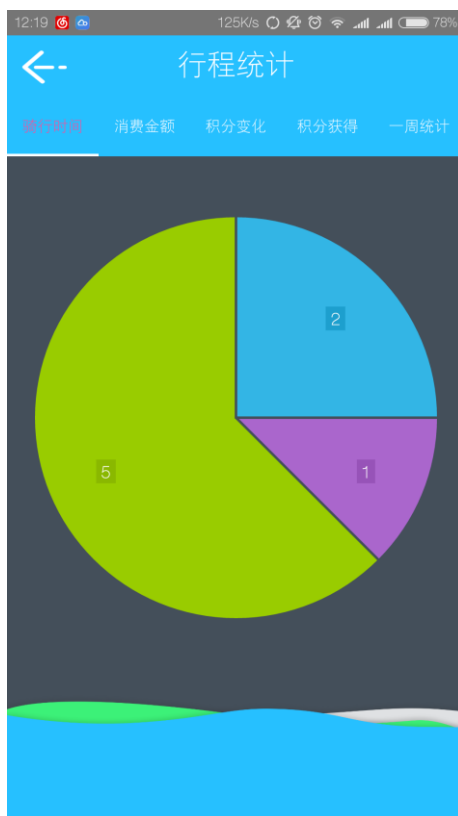


图 7. 行程统计界面

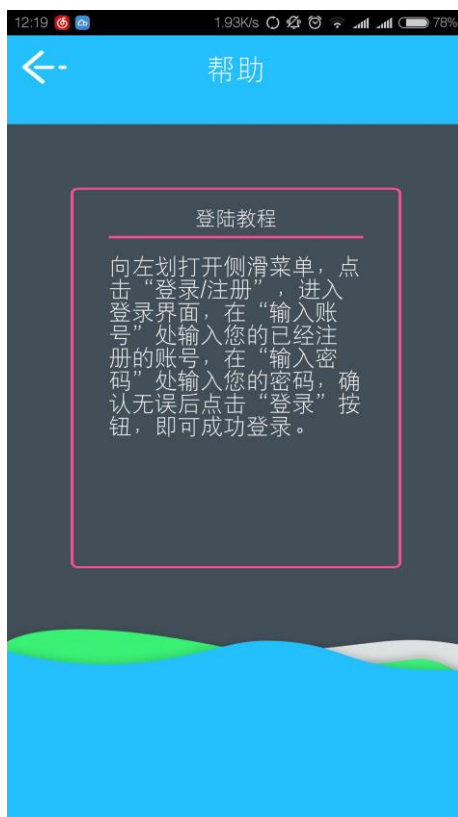


图 8. 帮助界面

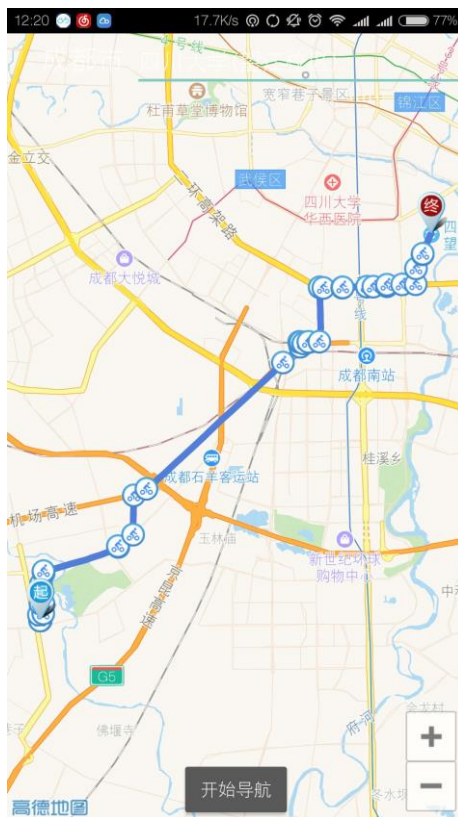


图 9. 导航界面



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